Advances in Intelligent Systems and Computing 287

Tutut Herawan
Rozaida Ghazali
Mustafa Mat Deris *Editors*

Recent Advances on Soft Computing and Data Mining

Proceedings of the First International Conference on Soft Computing and Data Mining (SCDM-2014) Universiti Tun Hussein Onn Malaysia Johor, Malaysia June 16th–18th, 2014

Springer

Advances in Intelligent Systems and Computing

Volume 287

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

For further volumes:
http://www.springer.com/series/11156

About this Series

The series “Advances in Intelligent Systems and Computing” contains publications on theory, applications, and design methods of Intelligent Systems and Intelligent Computing. Virtually all disciplines such as engineering, natural sciences, computer and information science, ICT, economics, business, e-commerce, environment, healthcare, life science are covered. The list of topics spans all the areas of modern intelligent systems and computing.

The publications within “Advances in Intelligent Systems and Computing” are primarily textbooks and proceedings of important conferences, symposia and congresses. They cover significant recent developments in the field, both of a foundational and applicable character. An important characteristic feature of the series is the short publication time and world-wide distribution. This permits a rapid and broad dissemination of research results.

Tutut Herawan · Rozaida Ghazali
Mustafa Mat Deris
Editors

Recent Advances on Soft Computing and Data Mining

Proceedings of the First International Conference on Soft Computing and Data Mining (SCDM-2014)
Universiti Tun Hussein Onn Malaysia, Johor, Malaysia June, 16th–18th, 2014

Springer

Editors
Tutut Herawan
Faculty of Computer Science and
Information Technology
University of Malaya
Kuala Lumpur
Malaysia

Mustafa Mat Deris
Faculty of Computer Science and
Information Technology
Universiti Tun Hussein Onn Malaysia
Malaysia

Rozaida Ghazali
Faculty of Computer Science and
Information Technology
Universiti Tun Hussein Onn Malaysia
Malaysia

ISSN 2194-5357 ISSN 2194-5365 (electronic)
ISBN 978-3-319-07691-1 ISBN 978-3-319-07692-8 (eBook)
DOI 10.1007/978-3-319-07692-8
Springer Cham Heidelberg New York Dordrecht London

Library of Congress Control Number: 2014940281

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

We are honored to be part of this special event in the First International Conference on Soft Computing and Data Mining (SCDM-2014). SCDM-2014 will be held at Universiti Tun Hussein Onn Malaysia, Johor, Malaysia on June 16th–18th, 2014. It has attracted 145 papers from 16 countries from all over the world. Each paper was peer reviewed by at least two members of the Program Committee. Finally, only 65 (44%) papers with the highest quality were accepted for oral presentation and publication in these volume proceedings.

The papers in these proceedings are grouped into two sections and two in conjunction workshops:

- Soft Computing
- Data Mining
- Workshop on Nature Inspired Computing and Its Applications
- Workshop on Machine Learning for Big Data Computing

On behalf of SCDM-2014, we would like to express our highest gratitude to be given the chance to cooperate with Applied Mathematics and Computer Science Research Centre, Indonesia and Software and Multimedia Centre, Universiti Tun Hussein Onn Malaysia for their support. Our special thanks go to the Vice Chancellor of Universiti Tun Hussein Onn Malaysia, Steering Committee, General Chairs, Program Committee Chairs, Organizing Chairs, Workshop Chairs, all Program and Reviewer Committee members for their valuable efforts in the review process that helped us to guarantee the highest quality of the selected papers for the conference.

We also would like to express our thanks to the four keynote speakers, Prof. Dr. Nikola Kasabov from KEDRI, Auckland University of Technology, New Zealand; Prof. Dr. Hamido Fujita from Iwate Prefectural University (IPU); Japan, Prof. Dr. Hojjat Adeli from The Ohio State University; and Prof. Dr. Mustafa Mat Deris from SCDM, Universiti Tun Hussein Onn Malaysia.

Our special thanks are due also to Prof. Dr. Janusz Kacprzyk and Dr. Thomas Ditzinger for publishing the proceeding in Advanced in Intelligent and Soft Computing of Springer. We wish to thank the members of the Organizing and Student Committees for their very substantial work, especially those who played essential roles.

We cordially thank all the authors for their valuable contributions and other participants of this conference. The conference would not have been possible without them.

Editors

Tutut Herawan
Rozaida Ghazali
Mustafa Mat Deris

Conference Organization

Patron

Prof. Dato' Dr. Mohd Noh Bin Dalimin	Vice-Chancellor of Universiti Tun Hussein Onn Malaysia

Honorary Chair

Witold Pedrycz	University of Alberta, Canada
Junzo Watada	Waseda University, Japan
Ajith Abraham	Machine Intelligence Research Labs, USA
A. Fazel Famili	National Research Council of Canada
Hamido Fujita	Iwate Prefectural University, Japan

Steering Committee

Nazri Mohd Nawi	Universiti Tun Hussein Onn Malaysia, UTHM
Jemal H. Abawajy	Deakin University, Australia

Chair

Rozaida Ghazali	Universiti Tun Hussein Onn Malaysia
Tutut Herawan	Universiti Malaya
Mustafa Mat Deris	Universiti Tun Hussein Onn Malaysia

Secretary

Noraini Ibrahim	Universiti Tun Hussein Onn Malaysia
Norhalina Senan	Universiti Tun Hussein Onn Malaysia

Organizing Committee

Hairulnizam Mahdin	Universiti Tun Hussein Onn Malaysia
Suriawati Suparjoh	Universiti Tun Hussein Onn Malaysia

Rosziati Ibrahim	Universiti Tun Hussein Onn Malaysia
Mohd. Hatta b. Mohd. Ali @ Md. Hani	Universiti Tun Hussein Onn Malaysia
Nureize Arbaiy	Universiti Tun Hussein Onn Malaysia
Noorhaniza Wahid	Universiti Tun Hussein Onn Malaysia
Mohd Najib Mohd Salleh	Universiti Tun Hussein Onn Malaysia
Rathiah Hashim	Universiti Tun Hussein Onn Malaysia

Program Committee Chair

Mohd Farhan Md Fudzee	Universiti Tun Hussein Onn Malaysia
Shahreen Kassim	Universiti Tun Hussein Onn Malaysia

Proceeding Chair

Tutut Herawan	Universiti Malaya
Rozaida Ghazali	Universiti Tun Hussein Onn Malaysia
Mustafa Mat Deris	Universiti Tun Hussein Onn Malaysia

Workshop Chair

Prima Vitasari	Institut Teknologi Nasional, Indonesia
Noraziah Ahmad	Universiti Malaysia Pahang

Program Committee

Soft Computing

Abir Jaafar Hussain	Liverpool John Moores University, UK
Adel Al-Jumaily	University of Technology, Sydney
Ali Selamat	Universiti Teknologi Malaysia
Anca Ralescu	University of Cincinnati, USA
Azizul Azhar Ramli	Universiti Tun Hussein Onn Malaysia
Dariusz Krol	Wroclaw University, Poland
Dhiya Al-Jumeily	Liverpool John Moores University, UK
Ian M. Thornton	University of Swansea, UK
Iwan Tri Riyadi Yanto	Universitas Ahmad Dahlan, Indonesia
Jan Platos	VSB-Technical University of Ostrava
Jon Timmis	University of York Heslington, UK
Kai Meng Tay	UNIMAS
Lim Chee Peng	Deakin University
Ma Xiuqin	Northwest Normal University, PR China
Mamta Rani	Krishna Engineering College, India

Meghana R. Ransing	University of Swansea, UK
Muh Fadel Jamil Klaib	Jadara University, Jordan
Mohd Najib Mohd Salleh	Universiti Tun Hussein Onn Malaysia
Mustafa Mat Deris	Universiti Tun Hussein Onn Malaysia
Natthakan Iam-On	Mae Fah Luang University, Thailand
Nazri Mohd Nawi	Universiti Tun Hussein Onn Malaysia
Qin Hongwu	Northwest Normal University, PR China
R.B. Fajriya Hakim	Universitas Islam Indonesia
Rajesh S. Ransing	University of Swansea, UK
Richard Jensen	Aberystwyth University
Rosziati Ibrahim	Universiti Tun Hussein Onn Malaysia
Rozaida Ghazali	Universiti Tun Hussein Onn Malaysia
Russel Pears	Auckland University of Technology
Safaai Deris	Universiti Teknologi Malaysia
Salwani Abdullah	Universiti Kebangsaan Malaysia
Shamshul Bahar Yaakob	UNIMAP
Siti Mariyam Shamsuddin	Universiti Teknologi Malaysia
Siti Zaiton M. Hashim	Universiti Teknologi Malaysia
Theresa Beaubouef	Southeastern Louisiana University
Tutut Herawan	Universiti Malaya
Yusuke Nojima	Osaka Prefecture University

Data Mining

Ali Mamat	Universiti Putra Malaysia
Bac Le	University of Science, Ho Chi Minh City, Vietnam
Bay Vo	Ho Chi Minh City University of Technology, Vietnam
Beniamino Murgante	University of Basilicata, Italy
David Taniar	Monash University
Eric Pardede	La Trobe University
George Coghill	University of Auckland
Hamidah Ibrahim	Universiti Putra Malaysia
Ildar Batyrshin	Mexican Petroleum Institute
Jemal H. Abawajy	Deakin University
Kamaruddin Malik Mohamad	Universiti Tun Hussein Onn Malaysia
La Mei Yan	ZhuZhou Institute of Technology, PR China
Md Anisur Rahman	Charles Sturt University, Australia
Md Yazid Md Saman	Universiti Malaysia Terengganu
Mohd Hasan Selamat	Universiti Putra Malaysia
Naoki Fukuta	Shizuoka University
Noraziah Ahmad	Universiti Malaysia Pahang
Norwati Mustapha	Universiti Putra Malaysia
Patrice Boursier	University of La Rochelle, France

Prabhat K. Mahanti	University of New Brunswick, Canada
Roslina Mohd Sidek	Universiti Malaysia Pahang
Palaiahnakote Shivakumara	Universiti Malaya
Patricia Anthony	Lincoln University, New Zealand
Sofian Maabout	Université Bordeaux, France
Shuliang Wang	Wuhan University
Tetsuya Yoshida	Hokkaido University
Vera Yuk Ying Chung	University of Sydney
Wan Maseri Wan Mohd	Universiti Malaysia Pahang
Wenny Rahayu	La Trobe University
Yingjie Hu	Auckland University of Technology
You Wei Yuan	ZhuZhou Institute of Technology, PR China
Zailani Abdullah	Universiti Malaysia Terengganu

Workshop on Nature Inspired Computing and Its Applications

Somnuk Phon-Amnuaisuk (Chair)	Institut Teknologi Brunei
Adham Atyabi	Flinders University
Ak Hj Azhan Pg Hj Ahmad	Institut Teknologi Brunei
Atikom Ruekbutra Mahanakorn	University of Technology
Au Thien Wan	Institut Teknologi Brunei
Hj Idham M. Hj Mashud	Institut Teknologi Brunei
Hj Rudy Erwan bin Hj Ramlie	Institut Teknologi Brunei
Ibrahim Edris	Institut Teknologi Brunei
Khor Kok Chin	Multimedia University
Ng Keng Hoong	Multimedia University
Somnuk Phon-Amnuaisuk	Institut Teknologi Brunei
Ting Choo Yee	Multimedia University
Werasak Kurutach	Mahanakorn University of Technology

Workshop on Machine Learning for Big Data Computing

Norbahiah Ahmad (Chair)	Universiti Teknologi Malaysia
Siti Mariyam Shamsuddin	Universiti Teknologi Malaysia

Contents

Soft Computing Track

Data Mining Track

Workshop on Nature Inspired Computing and Its Applications

Workshop on Machine Learning for Big Data Computing

A Fuzzy Time Series Model in Road Accidents Forecast

Lazim Abdullah and Chye Ling Gan

School of Informatics and Applied Mathematics,
Universiti Malaysia Terengganu,
21030 Kuala Terengganu, Malaysia
{lazim abdullah,lazim_m}@umt.edu.my

Abstract. Many researchers have explored fuzzy time series forecasting models with the purpose to improve accuracy. Recently, Liu et al., have proposed a new method, which an improved version of Hwang et al., method. The method has proposed several properties to improve the accuracy of forecast such as levels of window base, length of interval, degrees of membership values, and existence of outliers. Despite these improvements, far too little attention has been paid to real data applications. Based on these advantageous, this paper investigates the feasibility and performance of Liu et al., model to Malaysian road accidents data. Twenty eight years of road accidents data is employed as experimental datasets. The computational results of the model show that the performance measure of mean absolute forecasting error is less than 10 percent. Thus it would be suggested that the Liu et al., model practically fit with the Malaysian road accidents data.

Keywords: Fuzzy time series, time-variant forecast, length of interval, window base, road accidents.

1 Introduction

In the new global economy, forecasting plays important activities in daily lives as it often has been used to forecast weather, agriculture produce, stock price and students' enrolment. One of the most traditional approaches in forecasting is Box Jenkins model which was proposed by Box and Jenkin [1]. Traditional forecasting methods can be dealt with many forecasting cases. However, one of the limitations in implementing traditional forecasting is its incapability in fulfilling sufficient historical data. To solve this problem, Song and Chissom [2] proposed the concept of fuzzy time series. This concept was proposed when they attempted to implement fuzzy set theory for forecasting task. Over time, this method had been well received by researchers due to its capability in dealing with vague and incomplete data. Fuzzy set theory is created for handling of uncertain environment and fuzzy numbers provided various opportunities to compile difficult and complex problems. Fuzzy time series method is appeared successfully dealt with uncertainty of series and in some empirical studies, it provides higher accuracy. Some of the recent research in fuzzy time series performances can be retrieved from [3], [4], [5]. However, the issues of accuracy and performance of fuzzy time series still very much debated.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_1, © Springer International Publishing Switzerland 2014

Defining window bases variable *w* is one the efforts to increase forecasting accuracy especially in time-variant fuzzy time series. Song and Chissom [6] have showed that the effect of forecasting result with the changes of *w*. Hwang et al., [7] used Song and Chissom's method as a basis to calculate the variations of the historical data and recognition of window base, but one level is extended to *w* level. Hwang et al., [7] forecasted results were better than those presented by Song and Chissom's method due to the fact that the proposed method simplifies the arithmetic operation process. However, the time-variant model of Hwang et al.,[7] was not withstand as time keeps moving on and on. Liu et al., [8] took an initiative to revise Hwang et al.'s model with the purpose to overcome several drawbacks. Among the drawbacks of Hwang et al., are the length of intervals and number of intervals. Huarng [9] and Huarng and Yu [10] argue that different lengths and numbers of intervals may affect the accuracy of forecast. Furthermore, Hwang et al., method did not provide suggestions with regard to the determination of window base. Also, they uniformly set 0.5 as the membership values in the fuzzy set, without giving variations in degree. With a good intention to improve these drawbacks, Liu et al., [8] proposed a new model with the goals to effectively determine the best interval length, level of window base and degrees of membership values. These moves surely targeted to increase the accuracy of forecasted values. Although the Liu et al.,'s method is considered as a excellent technique to increase accuracy but the method has never been conceptualized in real applications. So far, however, there have been little discussions about testing this improved fuzzy time series to road accidents data.

In road accident forecast, many researchers used traditional ARMA to correct the error terms. Gandhi and Hu [11], for example, used a differential equation model to represent the accident mechanism with time-varying parameters and an ARMA process of white noise is attached to model the equation error. Another example, is the combination of a regression model and ARIMA model presented by Van den Bossche et al., [12]. In Malaysia, Law et al., [13] made a projection of the vehicle ownership rate to the year 2010 and use this projection to predict the road accident death in 2010 by using an ARIMA model. The projection takes into account the changes in population and the vehicle ownership rate. The relationship between death rate and population, vehicle ownership rate were described utilizing transfer noise function in ARIMA analysis. Other than ARIMA, Chang [14] analyzed freeway accident frequencies using negative binomial regression versus artificial neural network. In line with well acceptance of fuzzy knowledge in forecasting research, Jilani and Burney [15] recently presented a new multivariate stochastic fuzzy forecasting model. These new methods were applied for forecasting total number of car road accidents casualties in Belgium using four secondary factors. However there have been no studies to bring the light of the relationship between fuzzy time series model and road accidents data. The recent discovery of an improved fuzzy time series motivates the need to explore the applicability of the time variant fuzzy time series to road accidents data. The present paper takes an initiative to implement the fuzzy time series in forecasting of Malaysian road accidents. Specifically this paper intends to test the variant time fuzzy time series Liu et al., [8] model to the Malaysian road accidents data.

This paper is organized as follows. Conceptual definitions of time-variant fuzzy time series, window bases and its affiliates are discussed in Section 2. The vigour of computational steps of road accidents data are elucidated in Section 3. The short conclusion finally presented in Section 4.

2 Preliminaries

The concept of fuzzy logic and fuzzy set theory were introduced to cope with the ambiguity and uncertainty of most of the real-world problems. Chissom [6] introduced the concept of fuzzy time series and since then a number of variants were published by many authors. The basic concepts of fuzzy set theory and fuzzy time series are given by Song and Chissom [2] and some of the essentials are reproduced to make the study self-contained. The basic concepts of fuzzy time series are explained by Definition 1 to Definition 4.

Definition 1. Y(t)(t =...,0,1,2,...), is a subset of R. Let Y(t) be the universe of discourse defined by the fuzzy set $\mu_i(t)$ .If F(t) consists of $\mu_i(t)$ (i =1,2,...), F(t) is called a fuzzy time series on Y(t), i=1,2,…

Definition 2. If there exists a fuzzy relationship R(t-1, t), such that F(t) =F(t-1) ◦R(t-1, t), where ◦ is an arithmetic operator, then F(t) is said to be caused by F(t-1). The relationship between F(t) and F(t-1) can be denoted by F(t-1) →F(t).

Definition 3. Suppose F(t) is calculated by F(t-1) only, and F(t) = F(t-1) ◦R(t-1, t). For any t, if R(t-1, t) is independent of t, then F(t) is considered a time-invariant fuzzy time series. Otherwise, F(t) is time-variant.

Definition 4. Suppose F(t-1) = $\tilde{A}_i$ and F(t) = $\tilde{A}_j$, a fuzzy logical relationship can be defined as $\tilde{A}_i \rightarrow \tilde{A}_j$ where $\tilde{A}_i$ and $\tilde{A}_j$ are called the left-hand side and right-hand side of the fuzzy logical relationship, respectively.

These definitions become the basis in explaining fuzzy time series Liu et al., method. Liu et al., proposed the forecasting method with the aim at improving Hwang et al.'s method. Liu et al., method's has successfully overcome some drawbacks of Hwang et al., method's by finding the best combination between the length of intervals and the window bases. Detailed algorithms of Liu et al. [8], are not explained in this paper.

3 Implementation

In this experiment, Liu et al.,'s method is tested to the Malaysian road accidents data. An official road accidents data released by Royal Malaysian Police [16] are employed to the model. The calculation is executed in accordance with the proposed method. For the purpose of clarity and simplicity, the following computations are limited to forecasted value of the year 2009. Also, due to space limitation, the historical data from the year 2004 to 2008 are accounted in these computational steps.

Step 1: Collect the historical data of road accident in Malaysia., Dv_t, for the year 2004 to 2008.

Step 2: Examine outliers. The studentized residual analysis method is applied to determine whether there exist outliers in historical data. Statistical software is used to calculate the residual. Table 1 shows the outliers examination for the last five years before 2009.

Table 1. Studentized deleted residual of the historical data

Year	Number of Road Accident, Dv_t	Studentized deleted residual
2004	326,815	1.0272
2005	328,264	0.60248
2006	341,252	0.62921
2007	363,319	1.01975
2008	373,047	0.91899

It shows that all of studentized deleted residuals| are less than 2.5, thus confirm that there are no outliers in the historical data.

Step 3: Calculate the variation of the historical data. For example, the variation of year 2005 is calculated as follow:

$$\text{Variation} = Rv_{2005} - Rv_{2004} = 328{,}264 - 326{,}815 = 1{,}449$$

Similarly, the variations of all data are computed. It can be seen that the minimum of the variations in the data is -4,595 (D_{min}) and the maximum is 28,162 (D_{max}). To simplify computations, let D_1=405 and D_2=338.

$$U=[D_{min} - D_1 , D_{max} + D_2] =[-4{,}595-405, 28{,}162+338]=[-5{,}000, 28{,}500]$$

Step 4: Calculate *Ad* by dividing all the variations (Step 3) with number of data minus one:

$$Ad = \text{int}\left(\frac{10{,}904 + 5054 ... + 9{,}728}{29-1}\right) = \left(\frac{33{,}6663}{28}\right) = 12{,}023.68 \approx 12{,}200$$

For simplicity, *Ad*=12,200 is divided by 10 that yields the unit value 1,220. Thus, there are 10 possible interval lengths (1,220, 2,440, …, 12,200). The membership function for l=1,220 is 0.9. When l=2,440, its membership value is 0.8. The rest of membership function can be obtained in similar fashion. The corresponding relations are shown in Table 2.

Table 2. Interval length and the corresponding membership values for the road accident problem

Interval	Interval length	Membership value
1	1,220	0.9
2	2,440	0.8
3	3,660	0.7
4	4,880	0.6
5	6,100	0.5
6	7,320	0.4
7	8,540	0.3
8	9,760	0.2
9	10,980	0.1
10	12,200	0

Step 5: Using l=3,660 or membership value=0.7 as an example, the number of intervals (fuzzy set) is calculated as follows:

$$\text{Number of interval} = \frac{29{,}000-(-5{,}000)}{3{,}660} = 9.2896 \approx 10$$

Therefore, there are 10 intervals (fuzzy sets) and the interval midpoints are shown below.

u_1= [-5,000, -1340], Midpoint =-3,170
u_2= [-1,340, 2,320], Midpoint = 490
u_3= [2,320, 5,980], Midpoint =4,150
u_4= [5,980, 9,640], Midpoint =7,810
u_5= [9,640, 13,300], Midpoint =11,470
u_6= [13,300, 16,960], Midpoint =15,130
u_7= [16,960, 20,620], Midpoint =18,790
u_8= [20,620, 24,280], Midpoint =22,450
u_9= [24,280, 27,940], Midpoint =26,110
u_{10}= [27,940, 31,600], Midpoint =29,770

As shown in Table 2, when l=3,660, its membership value is 0.7. Thus, the fuzzy sets can be defined as follows.

$A_1=1/u_1+0.7/u_2+0/u_3+0/u_4+0/u_5+0/u_6+0/u_7+0/u_8+0/u_9+0/u_{10}$
$A_2=0.7/u_1+1/u_2+0.7/u_3+0/u_4+0/u_5+0/u_6+0/u_7+0/u_8+0/u_9+0/u_{10}$
$A_3=0/u_1+0.7/u_2+1/u_3+0.7/u_4+0/u_5+0/u_6+0/u_7+0/u_8+0/u_9+0/u_{10}$
$A_4=0/u_1+0/u_2+0.7/u_3+1/u_4+0.7/u_5+0/u_6+0/u_7+0/u_8+0/u_9+0/u_{10}$
$A_5=0/u_1+0/u_2+0/u_3+0.7/u_4+1/u_5+0.7/u_6+0/u_7+0/u_8+0/u_9+0/u_{10}$
$A_6=0/u_1+0/u_2+0/u_3+0/u_4+0.7/u_5+1/u_6+0.7/u_7+0/u_8+0/u_9+0/u_{10}$

$A_7=0/u_1+0/u_2+0/u_3+0/u_4+0/u_5+0.7/u_6+1/u_7+0.7/u_8+0/u_9+0/u_{10}$
$A_8=0/u_1+0/u_2+0/u_3+0/u_4+0/u_5+0/u_6+0.7/u_7+1/u_8+0.7/u_9+0/u_{10}$
$A_9=0/u_1+0/u_2+0/u_3+0/u_4+0/u_5+0/u_6+0/u_7+0.7/u_8+1/u_9+0.7/u_{10}$
$A_{10}=0/u_1+0/u_2+0/u_3+0/u_4+0/u_5+0/u_6+0/u_7+0/u_8+0.7/u_9+1/u_{10}$

Step 6: Fuzzify the variation of the data. If the variation at time i is within the scope of u_j, then it belongs to fuzzy set $\tilde{A}_j$. The fuzzy variation at time i is denoted as F (i). The variation between year 1991 and 1992 is 22,041, which falls in the range of u_8= [20,620, 24,280], so it belongs to the fuzzy set $\tilde{A}_8$. That is F(1992)= $\tilde{A}_8$. Similarly, the corresponding fuzzy sets of the remaining variations can be obtained.

Step 7: Calculate the fuzzy time series F(t) at window base *w*. The window base has to be more than or equal to 2 in order to perform a fuzzy composition operation. Therefore, w is set as 2 initially. Let C(t) be the criterion matrix of F(t) and $O^W(t)$ be the operation matrix at window base *w*.

The fuzzy relation matrix R(t) is computed by performing the fuzzy composition operation of C(t) and $O^W(t)$.

To get F(t), we can calculate the maximum of every column in matrix R(t).

Assume the window base is 4 and l=3,660. For example, the criterion matrix C(2009) of F(2009) is F(2008).

$$C(2009) = F(2008) = [\tilde{A}_5] = \begin{bmatrix} 0 & 0 & 0 & 0.7 & 1 & 0.7 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Meanwhile, the composition matrix is $O^4(2009)$ is composed of F(2007), F(2006) and F(2005).

$$O^4 = \begin{bmatrix} F(2007) \\ F(2006) \\ F(2005) \end{bmatrix} = \begin{bmatrix} \tilde{A}_8 \\ \tilde{A}_5 \\ \tilde{A}_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0.7 & 1 & 0.7 & 0 \\ 0 & 0 & 0 & 0.7 & 1 & 0.7 & 0 & 0 & 0 & 0 \\ 0.7 & 1 & 0.7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Apply the fuzzy composition operation to compute R(2009).

$$R(2009) = O^4(2009) \otimes C(2009)$$
$$= \begin{bmatrix} 0\times0 & 0\times0 & 0\times0 & 0\times0.7 & 0\times1 & 0\times0.7 & 0.7\times0 & 1\times0 & 0.7\times0 & 0\times0 \\ 0\times0 & 0\times0 & 0\times0 & 0.7\times0.7 & 1\times1 & 0.7\times0.7 & 0\times0 & 0\times0 & 0\times0 & 0\times0 \\ 0.7\times0 & 1\times0 & 0.7\times0 & 0\times0.7 & 0\times1 & 0\times0.7 & 0\times0 & 0\times0 & 0\times0 & 0\times0 \end{bmatrix}$$

Find the maximum at each column of R(2009) and F(2009) can be obtained as

$$\text{F(2009)}= \begin{bmatrix} 0 & 0 & 0 & 0.49 & 1 & 0.49 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Part of the results for F is given in Table 3

Table 3. Part of the fuzzy time series at window base *w*= 4

			F									
Year	Variation	Fuzzy Variation	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}
2004	28,162	$\tilde{A}_{10}$	0	0	0	0	0	1.4	1.4	0.49	0	0
2005	1,449	$\tilde{A}_2$	0	0	0	0	0	0	0	0	0	0
2006	12,988	$\tilde{A}_5$	0	0	0	0	0	0	0	0	0	0
2007	22,067	$\tilde{A}_8$	0	0	0	0	0	0.49	0	0	0	0
2008	9,728	$\tilde{A}_5$	0	0	0	0	0	0	0	0	0.49	0
2009			0	0	0	0.49	1	0.49	0	0	0	0

Step 8: In F(2009), there are three nonzero intervals: $u_{4,}$ $u_{5,}$ and u_6, while their interval midpoint are 7,810, 11,470, 15,130, individually. The computation of Cv_{2009} is as follows:

$$Cv_{2009} = \left[\frac{0.49 \times 7{,}810 + 1 \times 11{,}470 + 0.49 \times 15{,}130}{3} \right] = 7{,}570$$

Next, calculate the forecasted value of $Fv_{2009m,}$

$$Fv_{2009}=Cv_{2009}+Rv_{2008}=7{,}570+373{,}047=380{,}617$$

Step 9: To obtain the best forecasted values, the search algorithm is use to identify the best window base and interval length. Step 5 to step 8 are repeated for different window base and interval length. The best forecasted value is computed for different window base and interval length. Mean absolute deviation, MAD of each window base and interval length are calculated using the formula.

$$MAD = \frac{\sum_{t-1}^{n} | Fv_t - Rv_t |}{n - w - 1}$$

MAD values for each different window base and interval length are presented in Table 4.

Table 4. The value of MAD for different window base and interval length

	w=3	*w*=5	*w*=4	*w*=4	*w*=4	*w*=4
	i=0.6	i=0.6	i=0.5	i=0.6	i=0.7	i=0.8
Sum of $\|Fv_t\text{-}Rv_t\|$	257,473	239,312	223,234	226,262	204,781	222,879
MAD	10,619	10,878	9,706	9,837	8,904	9,690.4

Table 4 shows the lowest MAD happens when window base *w*=4 and interval length i=0.7. Window base *w*=4 and interval length i=0.7 are used to calculate the forecasted value for the year of 2009 using Step 7 and 8. It turns out that the forecasted value of road accident for the year 2009 is 380,617 cases.

The forecasted values for all the years are executed with the similar fashion. It involves huge computational loads therefore details of calculations for every tested year is not shown in this paper. To get better understanding of the forecasting performance, this paper provides trends for the forecasted results and the actual number of road accidents. The behaviours of these two curves can be seen in Fig 1.

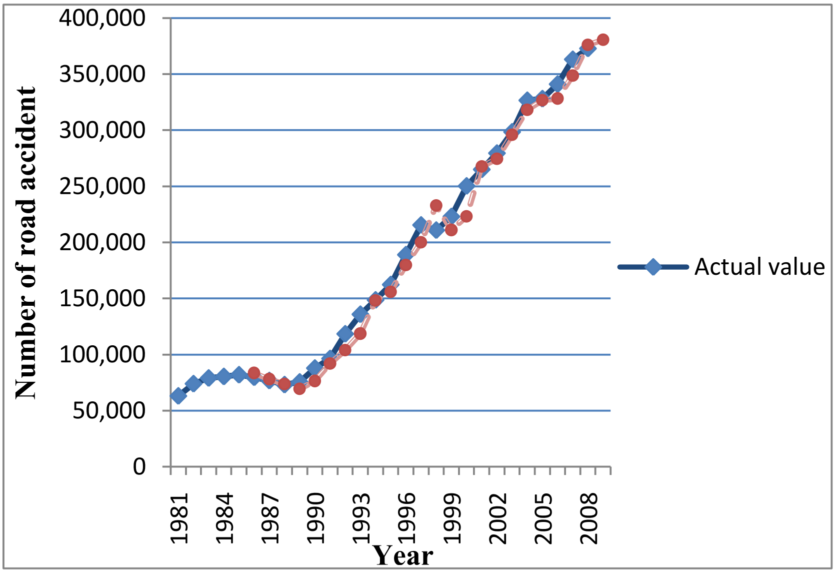


Fig. 1. Graph of actual versus forecasted value number of accidents

Performance of the model is also measured using mean absolute percentage error (MAPE, and mean square error (MSE). In the case of road accidents data, Liu et al., [5], method provides the three error measures as below.

MAPE (%) = 5.09
MSE = 106,670,326
MAD= 8,904

It is shows that Liu et al.,'s method gives less than 10 % of MAPE thus the Liu et al.,'s model can be considered as a good model in forecasting road accidents in Malaysia.

Summarily the method starts with examining studentized residual analysis method to check any outliers in the historical data. Upon calculation of the variations of historical data, the scope of the universe of discourse is defined, and the length of interval as well as its corresponding membership value is determined. A systematic search algorithm is then designed to determine the best combination between the length of intervals and window bases. The model performances are evaluated by observing error analysis with the actual data.

4 Conclusions

Fuzzy time series model has been widely used in forecasting with anticipation of finding viable results. Some researches state that fuzzy time series with refined model may gives more accurate forecasting result. Recently Liu et al., [8] proposed a refined model of Hwang et al., [7] and received much attention due to its capability of dealing with vague and incomplete data. This paper has initiated a move to test the capability of the time-variant fuzzy forecasting Liu et al.'s method to Malaysian road accidents data. The method has provided a systematic way to evaluate the length of intervals and the window base for road accident data. The model also considered outliers that normally influencing the overall performance of forecasting. The mean square error of less than 10 % validated the feasibility of the Liu et al.,'s model in forecasting of Malaysians road accidents data.

References

1. Box, G.E.P., Jenkins, G.: Time Series Analysis: Forecasting and Control. Holden-Day (1976)
2. Song, O., Chissom, B.S.: Forecasting enrolment with fuzzy time series, Part I. Fuzzy Sets Syst. 54, 1–10 (1993)
3. Abdullah, L.: Performance of exchange rate for ecast using distance-based fuzzy times series. Int. J. Eng. Tech. 5(1), 452–459 (2013)
4. Kamali, H.R., Shahnazari-Shahrezaei, P., Kazemipoor, H.: Two new time-variant methods for fuzzy time series forecasting. J. Intel. Fuzzy Syst. 24(4), 733–741 (2013)
5. Lee, M., Sadaei, H.: Improving TAIEX forecasting using fuzzy time series with box-cox power transformation. J. App. Stat. 40(11), 2407–2422 (2013)
6. Song, Q., Chissom, B.S.: Fuzzy forecasting enrollments with fuzzy time series, Part 2. Fuzzy Sets Syst. 62, 1–8 (1994)
7. Hwang, J.R., Chen, S.M., Lee, C.H.: Handling forecasting problems using fuzzy time series. Fuzzy Set Syst. 100, 217–228 (1998)
8. Liu, H.L., Wei, N.C., Yang, C.G.: Improved time-variant fuzzy time series forecast. Fuzzy Optim. Mak. 8, 45–65 (2009)
9. Huarng, K.H.: Effective lengths of intervals to improve forecasting in fuzzy time series. Fuzzy Set Syst. 123, 387–394 (2001)

10. Huarng, K.H., Yu, H.K.: A type 2 fuzzy time series model for stock index forecasting. Physica A: Statistical Mechanics and its Applications 353, 445–462 (2005)
11. Gandhi, U.N., Hu, S.J.: Data-based approach in modeling automobile crash. Int. J. Impact Eng. 16(1), 95–118 (1995)
12. Van den Bossche, F., Wets, G., Brijs, T.: A Regression Model with ARMA Errors to Investigate the Frequency and Severity of Road Traffic Accidents. In: Proceedings of the 83rd Annual Meeting of the Transportation Research Board, USA (2004)
13. Law, T.H., Radin Umar, R.S., Wong, S.V.: The Malaysian Government's Road Accident Death Reduction Target for Year 2010. Transportation Research 29(1), 42–50 (2004)
14. Chang, L.Y.: Analysis of freeway accident frequencies Negative binomial regression versus artificial neural network. Safety Sci. 43, 541–557 (2005)
15. Jilani, T.A., Burney, S.M.A., Ardil, C.: Multivariate High Order Fuzzy Time Series Forecasting for Car Road Accidents. World Academy of Science, Engineering and Technology 25, 288–293 (2007)
16. Royal Malaysia Police, Statistics of road accident and death, `http://www.rmp.gov.my/rmp` (accessed on April 8, 2010)

A Jordan Pi-Sigma Neural Network for Temperature Forecasting in Batu Pahat Region

Noor Aida Husaini[1], Rozaida Ghazali[1], Lokman Hakim Ismail[1], and Tutut Herawan[2,3]

[1] Universiti Tun Hussein Onn Malaysia
86400 Parit Raja, Batu Pahat, Johor, Malaysia
[2] University of Malaya
50603 Pantai Valley, Kuala Lumpur, Malaysia
[3] AMCS Research Center, Yogyakarta, Indonesia
gi090003@siswa.uthm.edu.my,
{rozaida,lokman}@uthm.edu.my, tutut@um.edu.my

Abstract. This paper disposes towards an idea to develop a new network model called a Jordan Pi-Sigma Neural Network (JPSN) to overcome the drawbacks of ordinary Multilayer Perceptron (MLP) whilst taking the advantages of Pi-Sigma Neural Network (PSNN). JPSN, a network model with a single layer of tuneable weights with a recurrent term added in the network, is trained using the standard backpropagation algorithm. The network was used to learn a set of historical temperature data of Batu Pahat region for five years (2005-2009), obtained from Malaysian Meteorological Department (MMD). JPSN's ability to predict the future trends of temperature was tested and compared to that of MLP and the standard PSNN. Simulation results proved that JPSN's forecast comparatively superior to MLP and PSNN models, with the combination of learning rate 0.1, momentum 0.2 and network architecture 4-2-1 and lower prediction error. Thus, revealing a great potential for JPSN as an alternative mechanism to both PSNN and MLP in predicting the temperature measurement for one-step-ahead.

Keywords: Jordan pi-sigma, Neural network, Temperature forecasting.

1 Introduction

Temperature has a significant impact on different sectors of activities which are exposed to temperature changes, agricultural interests and property [1]. One of the most sensitive issues in dealing with temperature forecasting is to consider that other variables might be affecting the temperature. Currently, temperature forecasting, which is a part of weather forecasting, is mainly issued in qualitative terms with the use of conventional methods, assisted by the data projected images taken by meteorological satellites to assess future trends [2, 3]. A great concern in developing methods for more accurate predictions for temperature forecasting employ the use of physical methods, statistical-empirical methods and numerical-statistical methods [4, 5]. Those methods for estimating temperature can work efficiently; however, it is

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_2, © Springer International Publishing Switzerland 2014

inadequate to represent the efficiency of temperature forecasting due to the relatively primitive output post-processing of the current techniques which is competitively superior to subjective prediction. Therefore, because temperature parameters itself can be nonlinear and complex, a powerful method is needed to deal with it [6].

With the advancement of computer technology and system theory, there have been more meteorological models conducted for temperature forecasting [2, 3], including soft computing approaches (e.g: neural network (NN), fuzzy systems, swarm techniques, etc.). For instance, Pal *et al.* [7] hybridised the MLP with Self-Organizing Feature Map (SOFM) to form a new model called SOFM-MLP to predict the maximum and minimum temperature by considering various atmospheric parameters. The SOFM-MLP was pre-processed by using Feature Selection (FS). They found that the combination of FS and SOFM-MLP produces good prediction by using only few atmospheric parameters as inputs. Similarly, Paras *et al.*, in their work [2] have discussed the effectiveness of NN with back-propagation learning to predict maximum temperature, minimum temperature and relative humidity. They observed that the NN becomes advantageous in predicting those atmospheric variables with high degree of accuracy, thus can be an alternative to traditional meteorological approaches. Lee, Wang & Chen [8] proposed a new method for temperature prediction to improve the rate of forecasting accuracy using high-order fuzzy logical relationships by adjusting the length of each interval in the universe of discourse. On the other hand, Smith, *et al.* [9] noted that ward-style NN can be used for predicting the temperature based on near real-time data with the reduction of prediction error by increasing the number of distinct observations in the training set.

Meanwhile, Radhika & Shashi [10] used Support Vector Machine (SVM) for one-step-ahead prediction. They found that SVM consistently gives better results compared to MLP trained with BP algorithm. Baboo & Shereef [11] forecast temperature using real-time dataset and compared it with practical working of meteorological department. Results showed that the convergence analysis is improved by using simplified Conjugate Gradient (CG) method. However, all of the studies mentioned above are considered as black box models, in which they take and give out information [2] without providing users with a function that describes the relationship between the input and output. Indeed, such approaches prone to overfit the data. Consequently, they also suffer long training times and often reach local minima in the error surface [12]. On the other hand, the development of Higher Order Neural Network (HONN) has captured researchers' attention. Pi-Sigma Neural Network (PSNN) which lies within this area, has the ability to converge faster and maintain the high learning capabilities of HONN [13]. The uses of PSNN itself for temperature forecasting are preferably acceptable. Yet, this paper focuses on developing a new alternative network model: Jordan Pi-Sigma Neural Network (JPSN) to overcome such drawbacks in MLP and taking the advantages of PSNN with the recurrent term added for temporal sequences of input-output mappings. Presently, JPSN is used to learn the historical temperature data of a suburban area in Batu Pahat, and to predict the temperature measurements for the next-day. These results might be helpful in modelling the temperature for predictive purposes.

The rest of this paper is organized as follow. Section 2 describes Jordan Pi-Sigma neural network. Section 3 describes experiments and comparison results. Finally, the conclusion of this work is described in Section 4.

2 Jordan Pi-Sigma Neural Network

This section discusses the motivations behind the development of JPSN, describes the basic architecture of JPSN, and outlines the learning processes of JPSN.

2.1 The Architecture of JPSN

The structure of JPSN is quite similar to the ordinary PSNN [14]. The main difference is the architecture of JPSN is constructed by having a recurrent link from output layer back to the input layer, just like the Jordan Neural Network (JNN) [15] had. Fig. 1 indicates the architecture of the proposed JPSN.

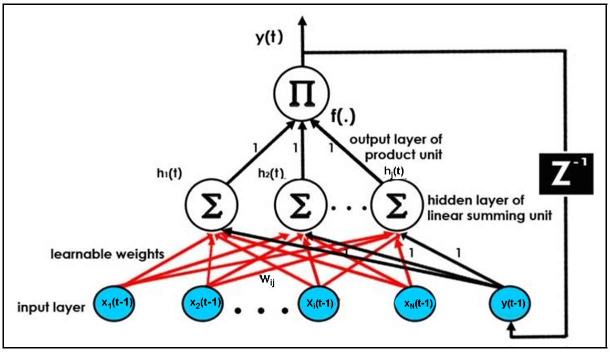


Fig. 1. The Architecture of JPSN

From Fig. 1, $x(t-1)$ is the i^{th} component of x at $(t-1)^{\text{th}}$ time, w_{ij} is the tuneable weights, j is the summing units, N is the number of input nodes and f is a suitable transfer function. Unlike the ordinary PSNN, the JPSN deals with a factor to reduce the current value in the context node $y(\mathrm{t}-1)$ before addressing the new copy value to the value in the context node. This feature provides the JPSN with storage capabilities by retaining previous output values in an attempt to model a memory of past event values. Those feedback connection results in nonlinear nature of the neuron. Typically, JPSN starts with a small network order for a given problem and then the recurrent link is added during the learning process. The learning process stops when the mean squared of the error is less than a pre-specified minimum error (0.0001). The new copy value can be examined by:

New copy value = Output Activation Value + Existing Copy Value * Weight Factor

Let the number of summing unit to be j and W to be the weight matrix of size $j \times (N+1)$, therefore, the overall input at time $(t-1)$ can be considered as $x_i(t-1) + y(t-1)$ whereas $i = 1, \ldots N$ and $y(t-1)$ is referred to $z(t-1)$. Since the JPSN are focusing on time-series forecasting, only one context unit node is needed. Weights from the input layers $x_i(t-1)$ to the summing units' j are tuneable, while weights between the summing unit layers and the output layer are fixed to 1. The tuned weights are used for network testing to see how well the network model generalises on unseen data. Z^{-1} denotes time delay operation. $y(t)$ indicates the output of the k^{th} node in the $(t-1)^{\text{th}}$ time, which is employed as a new input of the i^{th} layer. The context unit node influenced the input node which represents the vital of JPSN model. Weights from the content unit node $y(t-1)$ to the summing unit j are set to 1 in order to reduce the complexity of the network.

2.2 The Learning Algorithm of JPSN

We used back-propagation (BP) algorithm [16] with the recurrent link from output layer back to the input layer nodes for supervised learning in the JPSN. We initialised the weights to a small random value before adaptively trained the weights. Generally, JPSN can be operated in the following steps:

For each training example,

(1) Calculate the output.

$$y(t) = f\left(\prod_{j=1}^{M} h_j(t)\right), \tag{1}$$

where $h_j(t)$ can be calculated as:

$$\begin{aligned} h_j(t) &= \sum_{i=1}^{N+1} w_{ij} x_i(t-1) + w_{(N+1)j} z_i(t-1) \\ &= \sum_{i=1}^{N+1} w_{ij} z_i(t-1), \end{aligned} \tag{2}$$

where $h_j(t)$ represents the activation of the j unit at time t. The unit's transfer function f sigmoid activation function, which bounded the output range into the of $[0,1]$.

(2) Compute the output error at time (t) using standard Mean Squared Error (MSE) by minimising the following index:

$$E = \frac{1}{n_{tr}} \sum^{n_{tr}} [d(t) - y(t)]^2, \tag{3}$$

where n_{tr} denotes the number of training sets. This step is completed repeatedly for all nodes in the current layer at time step $(t-1)$.

(3) By adapting the BP gradient descent algorithm, compute the weight changes by:

$$\Delta w_{ij} = \eta \left(\prod_{j=1}^{M} h_j \right) x_i , \tag{4}$$

where h_j is the output of summing unit and η is the learning rate. The learning rate is used to control the learning step, and has a very important effect on convergence time.

(4) Update the weight:

$$w_{ij} = w_{ij} + \Delta w_{ij} . \tag{5}$$

(5) To accelerate the convergence of the error in the learning process, the momentum term, α is added into Equation (5). Then, the values of the weight for the interconnection on neurons are calculated and can be numerically expressed as

$$w_{ij} = w_{ij} + \alpha \Delta w_{ij} , \tag{6}$$

where the value of α is a user-selected positive constant $(0 \le \alpha \le 1)$. The JPSN algorithm is terminated when all the stopping criteria (training error, maximum epoch and early stopping) are satisfied. If not, repeat step (1).

Algorithm 1. JPSN Algorithm

The utilization of product units in the output layer indirectly incorporates the capabilities of JPSN while using a small number of weights and processing units. Therefore, the proposed JPSN combines the properties of both PSNN and JNN so that better performance can be achieved. When utilising the proposed JPSN as predictor for one-step-ahead, the previous input values are used to predict the next element in the data. The unique architecture of JPSN may also avoid from the combinatorial explosion of higher-order terms as the network order increases. The JPSN has a topology of a fully connected two-layered feedforward network. Considering the fixed weights that are not tuneable, it can be said that the summing layer is not "hidden" as in the case of the MLP. This is by means; such a network topology with only one layer of tuneable weights may reduce the training time.

3 Experiments Results and Discussion

In this section, we implemented JPSN using MATLAB 7.10.0 (R2010a) on Pentium® Core $^{\text{TM}2}$ Quad CPU. All the networks were trained and tested with daily temperature data gathered from National Forecast Office, Malaysian Meteorological Department (MMD). The network models were built considering five (5) different numbers of input nodes ranging from 4 to 8 [17]. A single neuron was considered for the output layer. The number of hidden layer (for MLP), and higher order terms (for PSNN and JPSN) was initially started with 2 nodes, and increased by one until a maximum of 5 nodes [13, 18]. The combination of 4 to 8 input nodes and 2 to 5 nodes for hidden layer/higher order terms of the three (3) network models yields a total of 1215 Neural

Network (NN) architectures for each in-sample training dataset. Since the forecasting horizon is one-step-ahead, the output variable represents the temperature measurement of one-day ahead. Each of data series is segregated in time order and is divided into 3 sets; the training, validation and the out-of-sample data. To avoid computational problems, the data is normalised between the upper and lower bounds of the network transfer function, $1/\left(1+e^{-x}\right)$ [18].

3.1 Benchmarks

To test the performance of the 3 network models, we used several criteria which are Mean Squared Error (MSE), Signal to Noise Ratio (SNR) [19], Mean Absolute Error (MAE) [20] and Normalised Mean Squared Error (NMSE) [19]. The NMSE acts as an overall measure of bias and scatter, and also for measuring network performance. Obviously, the smaller NMSE value might give better performance. For the purpose of comparison, we used the following notations (refer to Table 1):

Table 1. Performance Metrics

Formulae	
$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(P_i - P_i^*\right)^2$	(7)
$NMSE = \frac{1}{\sigma^2 n}\sum_{i=1}^{n}\left(P_i - P_i^*\right)^2$ $\sigma^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left(P_i - P_i^*\right)^2$ $P_i^* = \sum_{i=1}^{n} P_i$	(8)
$SNR = 10 * \lg\left(\frac{m^2 * n}{SSE}\right)$ $SSE = \sum_{i=1}^{n}\left(P_i - P_i^*\right)$ $m = \max\left(P\right)$	(9)
$MAE = \frac{1}{n}\sum_{i=1}^{n}\lvert P_i - P_i^*\rvert$	(10)

where n is the total number of data patterns, P_i and P_i^* represent the actual and predicted output value.

3.2 Experimental Results

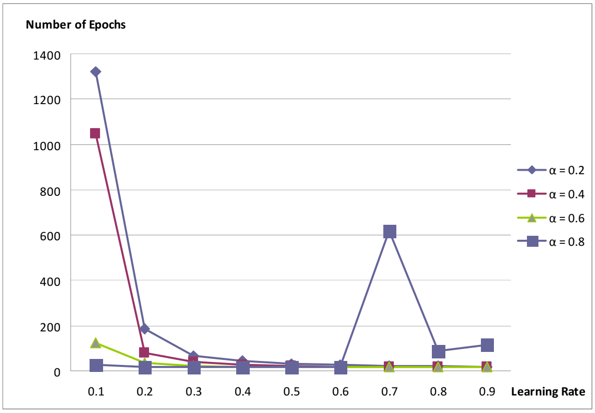
The temperature dataset collected from MMD was used to demonstrate the performance of the JPSN by considering a few different network parameters. Generally, the factors affecting the network performance include the learning factors, the higher order terms, and the number of neurons in the input layer. Extensive experiments have been conducted for training, testing and validation sets, and average results of 10 simulations/runs have been collected. The stopping criterion during the learning process was considered to be the pertinent epoch at the minimum error, which were set to 3000 and 0.0001 respectively [21]. To assess the performance of all network models used in this study, the aforementioned performance metrics in Section 3.1 are used. Convergence is achieved when the output of the network meets the earlier mentioned stopping criterion. Based on experiment, the best value for the momentum term $\alpha = 0.2$ and the learning rate $\eta = 0.1$, were chosen based on the simulation results being made by trial-and-error procedure. Likewise, number of input = 4 also have been chosen to be fixed in order to exemplify the effect of all network parameters.

3.2.1 The Network Parameters

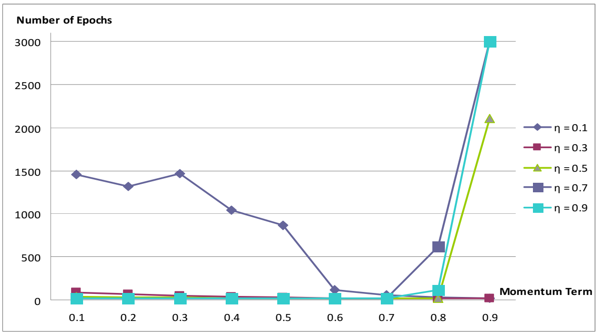
The network parameter, viz. the learning rate η and momentum term α are added in the training process. A higher **learning rate** can be used to speed up the learning process, however, if it is set too high, the algorithm might diverged and vice-versa. Fig. 2 (a) presents the number of epochs versus different values of learning rate with momentum; $\alpha = \{0.2, 0.4, 0.6, 0.8\}$. The figure clearly indicates that a higher learning rate leads the algorithm to converge quickly. It is however, the epochs start to rise when $\eta = 0.7$ and $\eta = 0.9$ (refer to $\alpha = 0.8$), in which the network began to overfit, thus leading to longer training time.

Furthermore, the **momentum term** is also an important factor for the learning process. Fig. 2 (b) indicates the effects of the momentum term on the model convergence with learning rate; $\eta = \{0.1, 0.3, 0.5, 0.7, 0.9\}$. In the early stage of training with small momentum, the number of epochs were reduced to some point, and then increased again. The number of epochs were increased when $\alpha \geq 0.7$ for most of learning rate, η. This is due to the smaller number of momentum, α which leads the network to diverge. Subsequently, it can be seen that larger value of momentum term affects the number of epochs reached. Therefore, the higher rate of momentum term can be used to achieve a smaller number of epochs. Thus, it can be concluded that a higher momentum term could give a positive catalyst for the network to converge. However, too large momentum value could also lead the network to easily get trapped into local minima. Moreover, one should consider the perfect combination of both learning rate and momentum term that might allow fast convergence rate. Therefore, instead of choosing the minimum epoch reached for each simulation, one should also consider the minimum error that obtained from the simulation process.

In this case, the combination of momentum term $\alpha = 0.2$ and the learning rate $\eta = 0.1$, were chosen for the rest of experimentations. The other combinations give smaller epochs but they consume higher error, which leads the network to get trapped into local minima. Therefore, such combinations were not considered to be used in this research. This can be seen on Fig. 3 which shows the Error VS Epochs for the combination of $\alpha = 0.8, \eta = 0.9$ and $\alpha = 0.2, \eta = 0.1$.

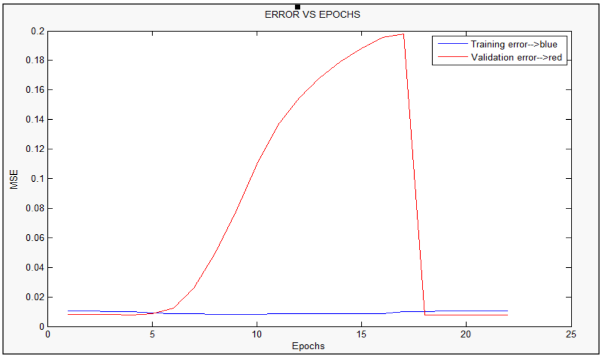


(a). Number of Epochs versus Various Learning Rate with Momentum 0.2, 0.4, 0.6 and 0.8

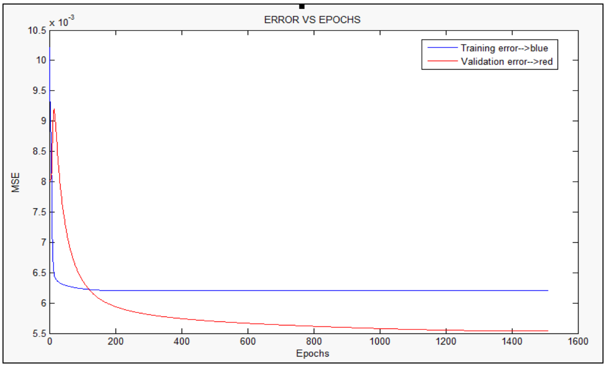


(b). Number of Epochs versus Various Momentum Term with Learning Rate 0.1, 0.3, 0.5, 0.7 and 0.9

Fig. 2. The Effects of Learning Factors on the Network Performance



(a). Error VS Epochs for momentum term $\alpha = 0.8$ and learning rate $\eta = 0.9$



(b). Error VS Epochs for momentum term $\alpha = 0.2$ and learning rate $\eta = 0.1$

Fig. 3. Error VS Epochs for the combination of $\alpha = 0.8, \eta = 0.9$ and $\alpha = 0.2, \eta = 0.1$

The number of **higher order terms** affects the learning capability and varies the complexity of the network structures. There is no upper limit to set the higher order terms. Yet, it is a rarely seen for the number of network order two times greater than the number of input nodes. Therefore, we gradually increased it, starting from 2nd order up to 5th order [13]. For the rest of the experiments, we used $\alpha = 0.2$, $\eta = 0.1$, as it has been proven for JPSN, they are the best parameters together with Input = 4. A comparative performance of the networks based on these parameter is tabularized in Table 2. The results apparently show that the JPSN with 2nd order outperformed the other network orders in terms of minimum error.

Table 2. The Effects of Number of Higher Order Terms for JPSN with $\alpha = 0.2$, $\eta = 0.1$ and Input = 4

ORDER	2	3	4	5
MAE	0.0635	0.0643	0.0646	0.0675
NMSE	0.7710	0.7928	0.8130	0.8885
SNR	18.7557	18.6410	18.5389	18.1574
MSE Training	0.0062	0.0064	0.0064	0.0076
MSE Testing	0.0065	0.0066	0.0068	0.0074
Epoch	1460.9	1641.1	1209.9	336.8

As there are no rigorous rules in the literature on how to determine the optimal **number of input neurons**, we used trial-and-error procedure between 4 and 8 to determine the numbers of input neurons. From Table 3, it can be observed that the network performance, on the whole, start to decrease (while the error starts to increase) when a larger number of input neurons is added. However, large number of neurons in the input layers is not always necessary, and it can decrease the network performance and may lead to greater execution time, which can caused overfitting.

Table 3. The Effects of the Input Neurons for JPSN with $\alpha = 0.2, \eta = 0.1$ and Input = 4

INPUT	4	5	6	7	8
MAE	0.0635	0.0632	0.0634	0.0632	0.0634
NMSE	0.7710	0.7837	0.7912	0.7888	0.8005
SNR	18.7557	18.6853	18.6504	18.6626	18.5946
MSE Training	0.0062	0.0062	0.0062	0.0062	0.0062
MSE Testing	0.0065	0.0066	0.0066	0.0066	0.0067
Epoch	1460.9	193.5	285.3	236.3	185.9

3.2.2 The Prediction of Temperature Measurement

The above discussions have shown that some network parameters may affect the network performances. In conjunction with that, it is necessary to illustrate the robustness of JPSN by comparing its performance with the ordinary PSNN and the MLP. Table 4 presents the best simulation results for JPSN, PSNN and MLP.

Table 4. Comparison of Results for JPSN, PSNN and MLP on All Measuring Criteria

Network Model	MAE	NMSE	SNR	MSE Training	MSE Testing	Epoch
JPSN	0.063458	0.771034	18.7557	0.006203	0.006462	1460.9
PSNN	0.063471	0.779118	18.71039	0.006205	0.006529	1211.8
MLP	0.063646	0.781514	18.69706	0.00623	0.006549	2849.9

Over all the training process, JPSN obtained the lowest MAE, which is 0.063458; while the MAE for PSNN and MLP were 0.063471 and 0.063646, respectively (refer to Fig. 4). By considering the MAE, it shows how close forecasts that have been made by JPSN are to the actual output in analysing the temperature. JPSN outperformed PSNN by ratio 1.95×10^{-4}, and 2.9×10^{-3} for the MLP.

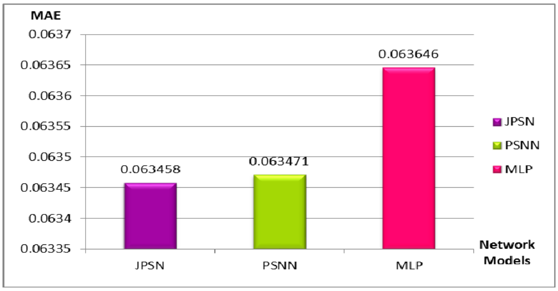


Fig. 4. MAE for JPSN, PSNN and MLP

Moreover, it can be seen that JPSN reached higher value of SNR (refer to Fig. 5). Therefore, it can be said that the network can track the signal better than PSNN and MLP. Apart from the MAE and SNR, it is verified that JPSN exhibited lower errors in both training and testing (refer to Fig. 6).

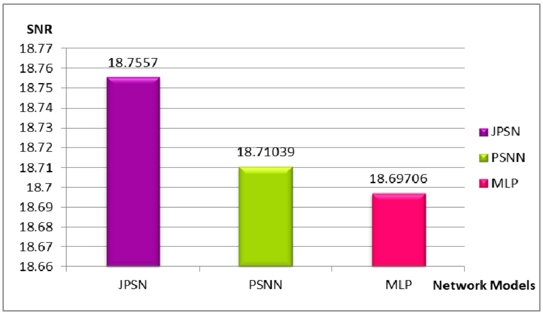


Fig. 5. SNR for JPSN, PSNN and MLP

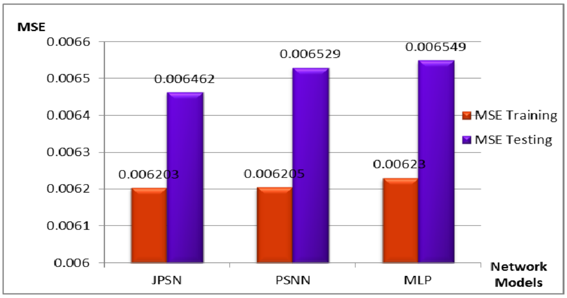


Fig. 6. MSE Training and Testing for JPSN, PSNN and MLP

The models' performances were also evaluated by comparing their NMSE. Fig. 7 illustrates the NMSE on the testing data set for the three network models. It shows that JPSN steadily gives lower NMSE when compared to both PSNN and MLP. This by means shows that the predicted and the actual values which were obtained by the JPSN are better than both comparable network models in terms of bias and scatter. Consequently, it can be inferred that the JPSN yield more accurate results, providing the choice of network parameters are determined properly. The parsimonious representation of higher order terms in JPSN assists the network to model successfully.

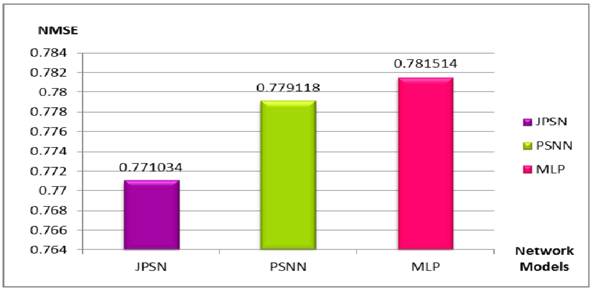


Fig. 7. NMSE for JPSN, PSNN and MLP

For purpose of demonstration, the earliest 10 data points (Day 1 to Day 10) were tabulated in Fig. 8, which indicate the predicted values (forecast) and the actual values (target) of temperature measurement for Batu Pahat region. Based on Fig. 8 (Day 1), the predicted error for JPSN is 0.4024 while for PSNN and MLP are 1.2404 and 1.2324, respectively. JPSN outperformed both network models by ratio :0.2449 for PSNN and ratio :0.2461 for MLP. For Day 2, still, JPSN leads PSNN and MLP with ratio :0.4446 and :0.4449, correspondingly. The rest of the comparisons in terms of the performance ratio is given in Table 5. From Table 5, it illustrates that JPSN exhibits minimum error for most of ten days compared to the two benchmarked models, PSNN and MLP.

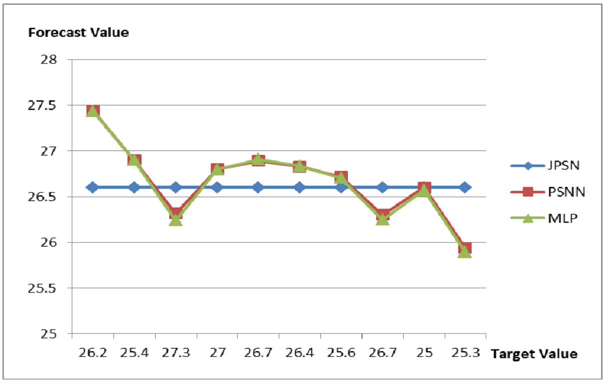


Fig. 8. Temperature Forecast made by JPSN, PSNN and MLP on 10 Data Points

Table 5. 10 Data Points of JPSN, PSNN and MLP Temperature Forecast

Forecast Value	Target Value (JPSN)	Target Value (PSNN)	Target Value (MLP)
26.2	26.6024	27.4404	27.4324
25.4	26.6034	26.9032	26.9014
27.3	26.6025	26.3165	26.2411
27.0	26.6011	26.8034	26.7992
26.7	26.601	26.8926	26.9105
26.4	26.602	26.8309	26.8391
25.6	26.6023	26.7183	26.7063
26.7	26.6016	26.3041	26.2483
25.0	26.6013	26.6013	26.57
25.3	26.6007	25.9385	25.8932

Fig. 9 represents the error minimisation by the three network models; JPSN, PSNN and MLP on 10 Data Points. Compared to the benchmarked models, the ordinary PSNN and the MLP, still, JPSN outperformed by having the least average error, 0.7006 compared to ordinary PSNN, 0.8301 and MLP, 0.8364.

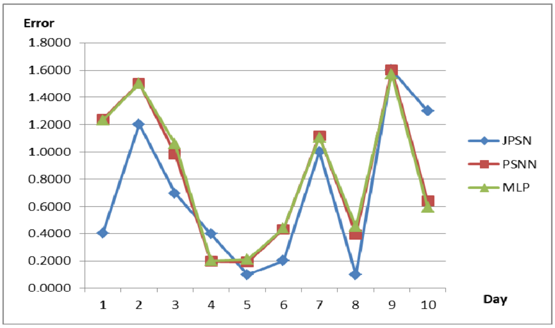


Fig. 9. Error Minimisation of JPSN, PSNN and MLP

On the whole, evaluations on MAE, NMSE, SNR, MSE Training and MSE Testing over the temperature data demonstrated that JPSN were merely improved the performance level compared to the two benchmarked network models, PSNN and MLP.

4 Conclusion

In this paper, we have presented a JPSN for temperature prediction for the next-day event. The 3 NN models: JPSN, PSNN and MLP were constructed, simulated and validated in 3 ways: (a) the minimum error in all performance metrics, (b) the effects of the network parameters and (c) the comparison done with the ordinary PSNN and the feedforward MLP. Overall, results clearly showed that the JPSN forecast defeated the ordinary PSNN and the feedforward MLP in terms of lower prediction error. Meanwhile, the prediction error that can be found in the JPSN on testing set is slightly lower that the other network architecture. This indicates that the JPSN is capable of representing nonlinear function. Consequently, it can be inferred that the JPSN yield more accurate results, providing the choice of network parameters are determined properly. From the extensive simulation results, it is proven that JPSN provides better prediction with **learning rate 0.1**, **momentum 0.2**, **network order 2**, and **number of input node 4**. Moreover, it should be noted that the comparison have been done among all these parameters. One of the basic practical aims of the present work was to examine whether the JPSN could be utilised as an alternative tool to the two benchmarked models. For the conclusion, the outcome shows that with a careful selection of network parameters, the performance of JPSN for temperature forecasting in Batu Pahat area can be improved.

Acknowledgments. This work is supported by University of Malaya High Impact Research Grant no vote UM.C/625/HIR/MOHE/SC/13/2 from Ministry of Higher Education Malaysia.

References

1. Baars, J.A., Mass, C.F.: Performance of National Weather Service Forecasts Compared to Operational, Consensus, and Weighted Model Output Statistics. Weather and Forecasting 20(6), 1034–1047 (2005)

2. Paras, et al.: A Feature Based Neural Network Model for Weather Forecasting. Proceedings of World Academy of Science, Engineering and Technology 34, 66–74 (2007)
3. Bhardwaj, R., et al.: Bias-free rainfall forecast and temperature trend-based temperature forecast using T-170 model output during the monsoon season 14, 351–360 (2007)
4. Barry, R., Chorley, R.: Chorley, Atmosphere, weather, and climate. Methuen (1982)
5. Lorenc, A.C.: Analysis methods for numerical weather prediction, vol. 112, pp. 1177–1194. John Wiley & Sons, Ltd. (1986)
6. Husaini, N.A., Ghazali, R., Nawi, N.M., Ismail, L.H.: Pi-Sigma Neural Network for Temperature Forecasting in Batu Pahat. In: Zain, J.M., Wan Mohd, W.M.b., El-Qawasmeh, E. (eds.) ICSECS 2011, Part II. CCIS, vol. 180, pp. 530–541. Springer, Heidelberg (2011)
7. Pal, N.R., et al.: SOFM-MLP: a hybrid neural network for atmospheric temperature prediction. IEEE Transactions on Geoscience and Remote Sensing 41(12), 2783–2791 (2003)
8. Lee, L.-W., Wang, L.-H., Chen, S.-M.: Temperature prediction and TAIFEX forecasting based on high-order fuzzy logical relationships and genetic simulated annealing techniques. Expert Systems with Applications 34(1), 328–336 (2008)
9. Smith, B.A., Hoogenboom, G., McClendon, R.W.: Artificial neural networks for automated year-round temperature prediction. Comput. Electron. Agric. 68(1), 52–61 (2009)
10. Radhika, Y., Shashi, M.: Atmospheric Temperature Prediction using Support Vector Machines. International Journal of Computer Theory and Engineering 1(1), 55–58 (2009)
11. Baboo, S.S., Shereef, I.K.: An Efficient Weather Forecasting System using Artificial Neural Network. International Journal of Environmental Science and Development 1(4), 321–326 (2010)
12. Yu, W.: Back Propagation Algorithm. Psychology/University of Newcastle (2005)
13. Ghazali, R., et al.: The application of ridge polynomial neural network to multi-step ahead financial time series prediction. Neural Computing & Applications 17, 311–323 (2008)
14. Shin, Y., Ghosh, J.: The Pi-Sigma Networks: An Efficient Higher-Order Neural Network for Pattern Classification and Function Approximation. In: Proceedings of International Joint Conference on Neural Networks, vol. 1, pp. 13–18 (1991)
15. Jordan, M.I.: Attractor dynamics and parallelism in a connectionist sequential machine. In: Proceedings of the Eighth Conference of the Cognitive Science Society, pp. 531–546 (1986)
16. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning Representations by Back-Propagating Errors. Nature 323(9), 533–536 (1986)
17. Lendasse, A., et al.: Non-linear Financial Time Series Forecasting - Application to the Bel 20 Stock Market Index. European Journal of Economic and Social Systems 14(1), 81–91 (2000)
18. Cybenko, G.: Approximation by Superpositions of a Sigmoidal Function. Signals Systems 2(303), 14 (1989)
19. Ghazali, R., Hussain, A., El-Dereby, W.: Application of Ridge Polynomial Neural Networks to Financial Time Series Prediction. In: International Joint Conference on Neural Networks (IJCNN 2006), pp. 913–920 (2006)
20. Valverde Ramírez, M.C., de Campos Velho, H.F., Ferreira, N.J.: Artificial neural network technique for rainfall forecasting applied to the São Paulo region. Journal of Hydrology 301(1-4), 146–162 (2005)
21. Rehman, M.Z., Nawi, N.M.: The Effect of Adaptive Momentum in Improving the Accuracy of Gradient Descent Back Propagation Algorithm on Classification Problems. Journal of Software Engineering and Computer Systems 179(6), 380–390 (2011)

A Legendre Approximation for Solving a Fuzzy Fractional Drug Transduction Model into the Bloodstream

Ali Ahmadian[1,*], Norazak Senu[1], Farhad Larki[2],
Soheil Salahshour[3], Mohamed Suleiman[1], and Md. Shabiul Islam[2]

[1] Institute for Mathematical Research (INSPEM), Universiti Putra Malaysia,
43400 UPM, Serdang, Selangor, Malaysia
[2] Institute of Microengineering and Nanoelectronics (IMEN), Universiti Kebangsan Malaysia (UKM), 43600 UKM, Bangi, Selangor, Malaysia
[3] Young Researchers and Elite Club, Mobarakeh Branch, Islamic Azad University,
Mobarakeh, Iran
{ahmadian.hosseini,farhad.larki}@gmail.com,
norazak,mohameds@upm.edu.my,
soheilsalahshour@yahoo.com

Abstract. While an increasing number of fractional order integrals and differential equations applications have been reported in the physics, signal processing, engineering and bioengineering literatures, little attention has been paid to this class of models in the pharmacokinetics-pharmacodynamic (PKPD) literature. In this research, we are confined with the application of Legendre operational matrix for solving fuzzy fractional differential equation arising in the drug delivery model into the bloodstream. The results illustrates the effectiveness of the method which can be in high agreement with the exact solution.

Keywords: Fuzzy fractional differential equations, Drug delivery process, Legendre polynomials, Operational matrix.

1 Introduction

The use of differential equations with non-integer powers of the differentiation order, namely fractional differential equations [1–3], is nowadays accepted in many fields of physics and engineering [4–6]. During these years, some authors have been studying various scientific computing approaches of fractional differential equations [7–9].

The reasons for the lack of application to pharmacodynamics are mainly two: first and foremost, PKPD models incorporating fractional calculus have not been proposed; second, it turns out that one needs to provide special algorithms to deal with such models. In this paper we present a family of PKPD models incorporating fuzzy fractional calculus, and investigate their behavior using purposely written algorithm. The main purpose of this paper is to attract the attention of the field into these possibly interesting family of PKPD models.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_3, © Springer International Publishing Switzerland 2014

Although some studies have been utilized to solve fractional differential equations analytically [10, 11], most nonlinear one does not have an exact analytical solution. Therefore, numerical and approximation techniques have been used for solving these equations [12–15].

As it is known, spectral methods play an important role in recent researches for numerical solution of differential equations in regular domains. It has been shown that spectral methods are powerful tools for solving various kinds of problems, due to their high accuracy. Doha et al. [16] proposed an efficient spectral tau and collocation methods based on Chebyshev polynomials for solving multi-term linear and nonlinear fractional differential equations subject to initial conditions. Furthermore, Bhrawy et al. [17] proved a new formula expressing explicitly any fractional-order derivatives of shifted Legendre polynomials of any degree in terms of shifted Legendre polynomials themselves, and the multi-order fractional differential equation with variable coefficients is treated using the shifted Legendre Gauss-Lobatto quadrature. Saadatmandi and Dehghan [18] and Doha et al. [19] derived the shifted Legendre and shifted Chebyshev operational matrices of fractional derivatives and used together spectral methods for solving fractional differential equations with initial and boundary conditions respectively.

On the other hand, after the time that Agarwal et al. [20] proposed the concept of solutions for fractional differential equations with uncertainty, a few researches have been done to find the fuzzy solution of FDEs by means of analytical and numerical methods [21–31]. The main motivation of this paper is to recommend a suitable way to approximate fuzzy fractional PKPD models using a shifted Legendre tau approach. This strategy demands a formula for fuzzy fractional-order Caputo derivatives of shifted Legendre polynomials of any degree which was provided by [26] and applied with tau method for solving fuzzy fractional PKPD with initial conditions.

2 Basic Concepts

In this section some preliminaries related to fuzzy fractional differential equations are provided. For more details see [2, 5, 23, 24, 32].

Definition 1. *Let u be a fuzzy set in $\mathbb{R}$. u is called a fuzzy number if:*
(i) u is normal: there exists $x_0 \in \mathbb{R}$ such that $u(x_0) = 1$;
(ii) u is convex: for all $x, y \in \mathbb{R}$ and $0 \leq \lambda \leq 1$, it holds that

$$u(\lambda x + (1 - \lambda)y \geq min\{u(x), u(y)\}$$

(iii) u is upper semi-continuous: for any $x_0 \in \mathbb{R}$, it holds that

$$u(x_0) \geq \lim_{x \to x_0^{\pm}} u(x).$$

(iv) $[u]^0 = \overline{\mathrm{supp(u)}}$ is a compact subset of $\mathbb{R}$.

In this paper, the set of all fuzzy numbers is denoted by $\mathbb{R}_{\mathcal{F}}$.

Definition 2. *([33]) The distance $D(u,v)$ between two fuzzy numbers u and v is defined as*

$$D(u,v)=\sup_{r\in[0,1]} d_H([u]^r,[v]^r),$$

where

$$d_H([u]^r,[v]^r)=\max\{|u_1^r-v_1^r|,|u_2^r-v_2^r|\},$$

is the Hausdorff distance between $[u]^r$ and $[v]^r$.

It is easy to see that D is a metric in $\mathbb{R}_{\mathcal{F}}$ and has the following properties (see [33], [34])

(i) $D(u\oplus w,v\oplus w)=D(u,v), \quad \forall u,v,w\in\mathbb{R}_{\mathcal{F}}$,
(ii) $D(k\odot u,k\odot v)=|k|D(u,v), \quad \forall k\in\mathbb{R},u,v\in\mathbb{R}_{\mathcal{F}}$,
(iii) $D(u\oplus v,w\oplus e)\leq D(u,w)+D(v,e), \quad \forall u,v,w\in\mathbb{R}_{\mathcal{F}}$,
(iv) $D(u+v,0)\leq D(u,0)+D(v,0), \ \forall u,v\in\mathbb{R}_{\mathcal{F}}$,
(v) $(\mathbb{R}_{\mathcal{F}},D)$ is a complete metric space.

Definition 3. *([35]) Let f and g be the two fuzzy-number-valued functions on the interval $[a,b]$, i.e., $f,g:[a,b]\to\mathbb{R}_{\mathcal{F}}$. The uniform distance between fuzzy-number-valued functions is defined by*

$$D^*(f,g):=\sup_{x\in[a,b]} D(f(x),g(x)) \tag{1}$$

Remark 1. ([35]) Let $f:[a,b]\to\mathbb{R}_{\mathcal{F}}$ be fuzzy continuous. Then from property (iv) of Hausdorff distance, we can define

$$D(f(x),\tilde{0})=\sup_{r\in[0,1]}\max\{|f_1^r(x)|,|f_2^r(x)|\}, \quad \forall x\in[a,b].$$

Theorem 1. *([36]) Let $F:(a,b)\to\mathbb{R}_{\mathcal{F}}$ be a function and denote $[F(t)]^r=[f_r(t),g_r(t)]$, for each $r\in[0,1]$. Then*
(1) If F is (1)-differentiable, then $f_r(t)$ and $g_r(t)$ are differentiable functions and

$$[F'(t)]^r=[f'_r(t),g'_r(t)],$$

(2) If F is (2)-differentiable, then $f_r(t)$ and $g_r(t)$ are differentiable functions and

$$[F'(t)]^r=[g'_r(t),f'_r(t)].$$

Definition 4. *([23]) Let $f:L^{\mathbb{R}_{\mathcal{F}}[a,b]}\cap C^{\mathbb{R}_{\mathcal{F}}}[a,b]$ and $x_0\in(a,b)$ and $\Phi(x)=\frac{1}{\Gamma(1-v)}\int_a^x\frac{f(t)}{(x-t)^v}dt$. We say that $f(x)$ is fuzzy Caputo fractional differentiable of order $0<v\leq 1$ at x_0, if there exists an element $({}^cD_{a^+}^v f)(x_0)\in C^{\mathbb{R}_{\mathcal{F}}[a,b]}[a,b]$ such that for all $0\leq r\leq 1$, $h>0$,*

$$(i) \qquad ({}^cD_{a^+}^v f)(x_0)=\lim_{h\to 0^+}\frac{\Phi(x_0+h)\ominus\Phi(x_0)}{h}=\lim_{h\to 0^+}\frac{\Phi(x_0)\ominus\Phi(x_0-h)}{h},$$

or

$$(ii) \qquad ({}^cD^v_{a+}f)(x_0) = \lim_{h\to 0^+} \frac{\Phi(x_0) \ominus \Phi(x_0+h)}{-h} = \lim_{h\to 0^+} \frac{\Phi(x_0-h) \ominus \Phi(x_0)}{-h},$$

or

$$(iii) \qquad ({}^cD^v_{a+}f)(x_0) = \lim_{h\to 0^+} \frac{\Phi(x_0+h) \ominus \Phi(x_0)}{h} = \lim_{h\to 0^+} \frac{\Phi(x_0-h) \ominus \Phi(x_0)}{-h},$$

or

$$(iv) \qquad ({}^cD^v_{a+}f)(x_0) = \lim_{h\to 0^+} \frac{\Phi(x_0) \ominus \Phi(x_0+h)}{-h} = \lim_{h\to 0^+} \frac{\Phi(x_0) \ominus \Phi(x_0-h)}{h}.$$

For sake of simplicity, we say that the fuzzy-valued function f ${}^c[(1)-v]$-differentiable if it is differentiable as in the Definition 4 case (i), and f is ${}^c[(2)-v]$-differentiable if it is differentiable as in the Definition 4 case(ii) and so on for the other cases.

The shifted Legendre polynomials, $L_n(x)$, are orthogonal with respect to the weight function $w_s(x) = 1$ in the interval $(0,1)$ with the orthogonality property:

$$\int_0^1 L_n(x)L_m(x)dx = \frac{1}{2n+1}\delta_{nm}. \tag{2}$$

The shifted Legendre polynomials are generated from the three-term recurrence relation:

$$\begin{array}{l} L_{i+1}(t) = \frac{(2i+1)(2x-1)}{i+1}L_i(x) - \frac{i}{i+1}L_{i-1}(x), \quad i = 1,2,... \\ L_0(x) = 1, \;\; L_1(x) = 2x-1. \end{array} \tag{3}$$

The analytic form of the shifted Legendre polynomial $L_n(x)$ of degree n is given by

$$L_n(x) = \sum_{i=0}^{n} (-1)^{n+i} \frac{(n+i)!}{(n-i)!} \frac{x^i}{(i!)^2} = \sum_{i=0}^{n} L_{i,n}x^i,$$

in which

$$L_{i,n} = (-1)^{n+i} \frac{(n+i)!}{(n-i)!(i!)^2},$$

where

$$L_n(0) = (-1)^n \;\; and \;\; L_n(1) = 1.$$

Definition 5. *[26]. For $y \in L_p^{\mathbb{E}}(0,1) \cap C^{\mathbb{E}}(0,1)$ and Legendre polynomial $L_n(x)$ a real valued function over $(0,1)$, the fuzzy function is approximated by*

$$y(x) = \sum_{j=0}^{\infty}{}^{*} c_j \odot L_j(x), \quad x \in (0,1),$$

where the fuzzy coefficients c_j are obtained by

$$c_j = (2j+1) \odot \int_0^1 y(x) \odot L_j(x)dx,$$

in which $L_j(x)$ is as the same in Eq. (3), and $\sum^*$ means addition with respect to $\oplus$ in $\mathbb{E}$.

3 Solution Method

We start by representing drug concentration in the effect compartment by the (Caputo) fractional differential equation:

$$^cD^\alpha y(t) + k_2 y(t) = k_1 A e^{-k_1 t} \quad y(0) = 0. \tag{4}$$

In the standard direct action model the effect at time t, $Y(t)$, is expressed by an arbitrary (memory-less) function of drug concentration in the effect site at time t, $G(y(t))$, however to generate a wider class of relationships, we assume that the effect at time $t \in [0, 1]$ is related to the fuzzy Caputo fractional derivative of $y(t)$. So we have

$$^cD^\alpha y(t) + k_2 y(t) = k_1 A e^{-k_1 t} \quad y(0; r) = [\underline{y}_0^r, \overline{y}_0^r], \tag{5}$$

in which $y(x) : L^{\mathbb{R}_\mathcal{F}}[0, 1] \cap C^{\mathbb{R}_\mathcal{F}}[0, 1]$ is a continuous fuzzy-valued function and $^cD^\alpha_{0+}$ denotes the fuzzy Caputo fractional derivative of order $\alpha \in [0, 1]$ and k_1, k_2 are positive coefficients. For more details see [29] and references there in.

For solving fuzzy fractional PKPD (5), we try to find a fuzzy function $y_m^{(r)} \in X_{\mathbb{R}_\mathcal{F}}$, therefore $(^cD^\alpha_{0+} y)(x)$, $y(x)$ and $f(x)$ be approximated using Definition 5 as:

$$y(x) \simeq \tilde{y}_m(x) = \sum_{j=0}^{m}{}^* c_j \odot L_j(x) = C_m^T \odot \varPhi_m, \tag{6}$$

that

$$[\tilde{y}_m(x)]^{(r)} = \sum_{j=0}^{m}{}^* c_j^{(r)} \odot [L_j(x)] \quad x \in I \subset R,$$

$$f(x) \simeq \tilde{f}_m(x) = \sum_{j=0}^{m}{}^* f_j \odot L_j(x) = F_m^T \odot \varPhi_m, \tag{7}$$

where $f(x) = k_1 A e^{-k_1 t}$ and $F_{m+1} = [f_0, f_1, ..., f_m]^T$ is obtained as

$$f_j = (2j+1) \odot \int_0^1 f(x) \odot L_j(x)dx. \tag{8}$$

Also, using relation (85) in [26] and Eq. (6) we obtain

$$ {}^{c}D^{\alpha}y(x) \simeq C^{T} \odot D^{\alpha}\varPhi_{m}(x) \simeq C^{T} \odot D^{(\alpha)}\varPhi_{m}(x). \tag{9} $$

Substituting Eqs. (6)-(9) in problem (5) and the coefficients $\left\{c_j^{(r)}\right\}_{j=0}^{m}$ are specified by imposing the equation to be almost fuzzy exact in Legendre operational matrix sense. Now, we establish fuzzy *residual* for the approximation of Eq. (5), when $[y(x)]^{(r)} \approx [\tilde{y}_m(x)]^{(r)}$.

It is expected that the deriving fuzzy function $[\tilde{y}_m(x)]^{(r)}$ will be a suitable approximation of the exact solution $[\tilde{y}(x)]^{(r)}$. To this end, let $X_{\mathbb{R}_{\mathcal{F}}} = L^2_{\mathbb{R}_{\mathcal{F}}}([0,1])$, let $\langle .,. \rangle_{\mathbb{R}_{\mathcal{F}}}$ indicate the fuzzy inner product for $X_{\mathbb{R}_{\mathcal{F}}}$. It is demanded that $R_m^{(r)}$ satisfy

$$ \left\langle R_m^{(r)}, L_i \right\rangle_{\mathbb{R}_{\mathcal{F}}} = \tilde{0}, \quad i = 0, 1, ..., m-1, \; r \in [0,1], \tag{10} $$

in which $\left\langle R_m^{(r)}, L_i \right\rangle_{\mathbb{R}_{\mathcal{F}}} = [(FR)\int_0^1 R_m(x) \odot L_i(x)dx]^{(r)}$. The left side is the shifted legendre coefficients associated with L_j. If $\{L_i\}_{i=0}^{m}$ are the main members of shifted Legendre family $\varPhi = \{L_i\}_{i=0}^{\infty}$ which is complete in $X_{\mathbb{R}_{\mathcal{F}}}$, then Eq. (10) needs the main terms to be zero in the Legendre extension of $R_m^{(r)}$ with respect to $\varPhi$ which is called tau method in crisp context and it is in a similar manner with the meaning of fuzzy . (for more details see [26, 37].

To discover $\tilde{y}_n^{(r)}$, implementing Eq.(10) to Eq.(5), with relation to Eqs. (6)-(9). We generate m fuzzy linear equations as

$$ \left\langle [R_m(x)]^{(r)}, L_i(x) \right\rangle = \left\langle [({}^{c}D^{\alpha}\tilde{y}_m)(x)]^{(r)} \ominus_g [(a_1) \odot \tilde{y}_m(x)]^{(r)} \ominus_g \left[(a_2) \odot \tilde{f}_m(x)\right]^{(r)}, L_i(x) \right\rangle = \tilde{0}, $$

for $i = 0, 1, ..., m-1$. Afterwards, substitution of Eq. (6) in the initial condition of Eq. (5) yields

$$ y(0) = \sum_{j=0}^{m}{}^{*} c_j^{(r)} \odot L_j(0) = y_0, $$

that this equation be coupled with the previous fuzzy linear equations and constructed $(m+1)$ fuzzy linear equations. Obviously, after solving this fuzzy system, the coefficients $\{c_j\}_{j=0}^{m}$ will be achieved.

4 Numerical Results

In this section the solution method which was presented in Section 3 is exploited for solving the fuzzy fractional drug transduction model (5). The results are confirmed the high accuracy of the method with only a few number of Legendre functions.

Now we reconsider the fuzzy fractional PKPD (5) as follows:

$$^{c}D^{\alpha}y(t)+k_2y(t)=k_1Ae^{-k_1t}\quad y(0;r)=[-1+r,1-r], \tag{11}$$

in which $y(x):L^{\mathbb{R}_{\mathcal{F}}}[0,1]\cap C^{\mathbb{R}_{\mathcal{F}}}[0,1]$ is a continuous fuzzy-valued function and $^{c}D^{\alpha}_{0+}$ denotes the fuzzy Caputo fractional derivative of order $\alpha\in[0,1]$. Also Let us consider from [29] that $k_2=0.0231$, but let k_1 vary (e.g., $0.6931, 0.11$, and 0.3) and $A=1$.

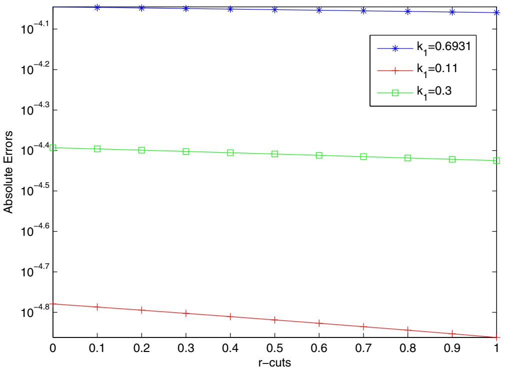


Fig. 1. The Absolute Errors of $\underline{\tilde{y}}^{(r)}_{n}$ for different k_1 with $m=8$ and $\alpha=0.85$

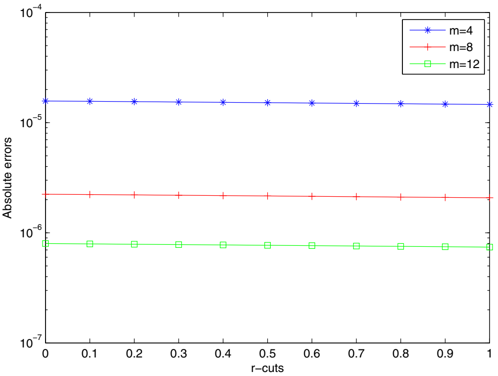


Fig. 2. The Absolute Errors for different values m with $\alpha=0.95$, $k_1=0.3$

The comparison between absolute errors of different k_1 obtained by the proposed method are shown in Fig. 1. The absolute error value for r-cut varied from 0 to 1 for different value of k_1 is calculated in Fig.1. As it can be observed at a constant r-cut by increasing the value of k_1 which is the ratio of variation of the drug in the bloodstream to the amount of drug in the GI-tract the value

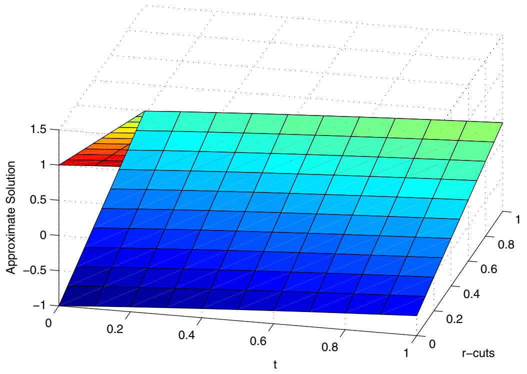


Fig. 3. The Fuzzy approximate solution with $k_1 = 0.3$, $\alpha = 0.95$ and $m = 8$

of absolute error increases. This is analogous to the previous reports for various values of the k_1 [38].

Moreover, different number of Legendre function has been experienced. According to the Fig. 2, one can find that with increasing the number of Legendre functions, the absolute errors is decreasing gradually. Finally, The approximate fuzzy solution by using the proposed method is shown in Fig. 3 to demonstrate the behavior of the solution as a fuzzy number in the time domain between 0 and 1.

5 Conclusion

In this paper, we introduce a numerical algorithm for solving fuzzy fractional kinetic model based on a Legendre tau method. The numerical example shows that our scheme could produce high accurate solution.

References

1. Oldham, K.B., Spainer, J.: The Fractional Calculus: Theory and Applications of Differentiation and Integration to Arbitrary Order. Academic Press, New York (1974)
2. Podlubny, I.: Fractional Differential Equations. Academic Press, San Diego (1999)
3. Kilbas, A.A., Srivastava, H.M., Trujillo, J.J.: Theory and Applications of Fractional Differential Equations. Elsevier, Amsterdam (2006)
4. Hilfer, R.: Application of Fractional Calculus in Physics. World Scientific, Singapore (2000)
5. Baleanu, D., Diethelm, K., Scalas, E., Trujillo, J.J.: Fractional Calculus Models and Numerical Methods. World Scientific Publishing Company (2012)
6. Golmankhaneh, A.K., Yengejeh, A.M., Baleanu, D.: On the fractional Hamilton and Lagrange mechanics. International Journal of Theoretical Physics 51, 2909–2916 (2012)

7. Lim, S.C., Eab, C.H., Mak, K.H., Li, M., Chen, S.Y.: Solving linear coupled fractional differential equations by direct operationalmethod and some applications. Mathematical Problems in Engineering 2012, 1–28 (2012)
8. Pedas, A., Tamme, E.: On the convergence of spline collocation methods for solving fractional differential equations. Journal of Computational and Applied Mathematics 235, 3502–3514 (2011)
9. Jiang, W.H.: Solvability for a coupled system of fractional differential equations at resonance. Nonlinear Analysis: Real World Applications 13, 2285–2292 (2012)
10. Jiang, H., Liu, F., Turner, I., Burrage, K.: Analytical solutions for the multi-term time-space Caputo–Riesz fractional advection–diffusion equationson a finite domain. J. Math. Anal. Appl. 389, 1117–1127 (2012)
11. Jiang, H., Liu, F., Turner, I., Burrage, K.: Analytical solutions for the generalized multi-term time-fractional diffusion-wave/diffusion equation in a finite domain. Comput. Math. Appl. 64, 3377–3388 (2012)
12. Momani, S., Shawagfeh, N.T.: Decomposition method for solving fractional Riccati differential equations. Appl. Math. Comput. 182, 1083–1092 (2006)
13. Song, L., Wang, W.: A new improved Adomian decomposition method and its application to fractional differential equations. Appl. Math. Model. 37, 1590–1598 (2013)
14. Odibat, Z., Momani, S., Xu, H.: A reliable algorithm of homotopy analysis method for solving nonlinear fractional differential equations. Appl. Math. Model. 34, 593–600 (2010)
15. Hashim, I., Abdulaziz, O., Momeni, S.: Homotopy analysis method for fractional IVPs. Commun. Nonlinear. Sci. Numer. Simul. 14, 674–684 (2009)
16. Doha, E.H., Bhrawy, A.H., Ezz-Eldien, S.S.: Efficient Chebyshev spectral methods for solving multi-term fractional orders differential equations. Appl. Math. Model. 35, 5662–5672 (2011)
17. Bhrawy, A.H., Alofi, A.S., Ezz-Eldien, S.S.: A quadrature tau method for variable coefficients fractional differential equations. Appl. Math. Lett. 24, 2146–2152 (2011)
18. Saadatmandi, A., Dehghan, M.: A new operational matrix for solving fractional-order differential equations. Comput. Math. Appl. 59, 1326–1336 (2010)
19. Doha, E.H., Bhrawy, A.H., Ezz-Eldien, S.S.: A Chebyshev spectral method based on operational matrix for initial and boundary value problems of fractional order. Comput. Math. Appl. 62, 2364–2373 (2011)
20. Agarwal, R.P., Lakshmikantham, V., Nieto, J.J.: On the concept of solution for fractional differential equations with uncertainty. Nonlinear Anal. 72, 2859–2862 (2010)
21. Allahviranloo, T., Salahshour, S., Abbasbandy, S.: Explicit solutions of fractional differential equations with uncertainty. Soft Comput. 16, 297–302 (2012)
22. Allahviranloo, T., Gouyandeh, Z., Armand, A.: Fuzzy fractional differential equations under generalized fuzzy Caputo derivative. J. Intelligent and Fuzzy Systems (2013), doi:10.3233/IFS-130831
23. Salahshour, S., Allahviranloo, T., Abbasbandy, S., Baleanu, D.: Existence and uniqueness results for fractional differential equations with uncertainty. Advances in Difference Equations 2012, 112 (2012)
24. Salahshour, S., Allahviranloo, T., Abbasbandy, S.: Solving fuzzy fractional differential equations by fuzzy Laplace transforms. Commun. Nonlinear. Sci. Numer. Simulat. 17, 1372–1381 (2012)
25. Ahmadian, A., Suleiman, M., Salahshour, S., Baleanu, D.: A Jacobi operational matrix for solving fuzzy linear fractional differential equation. Adv. Difference Equ. 2013, 104 (2013)

26. Ahmadian, A., Suleiman, M., Salahshour, S.: An Operational Matrix Based on Legendre Polynomials for Solving Fuzzy Fractional-Order Differential Equations. Abstract and Applied Analysis 2013, Article ID 505903, 29 pages (2013)
27. Balooch Shahriyar, M.R., Ismail, F., Aghabeigi, S., Ahmadian, A., Salahshour, S.: An Eigenvalue-Eigenvector Method for Solving a System of Fractional Differential Equations with Uncertainty. Mathematical Problems in Engineering 2013, Article ID 579761, 11 pages (2013)
28. Ghaemi, F., Yunus, R., Ahmadian, A., Salahshour, S., Suleiman, M., Faridah Saleh, S.: Application of Fuzzy Fractional Kinetic Equations to Modelling of the Acid Hydrolysis Reaction. Abstract and Applied Analysis 2013, Article ID 610314, 19 pages (2013)
29. Ahmadian, A., Senu, N., Larki, F., Salahshour, S., Suleiman, M., Shabiul Islam, M.: Numerical solution of fuzzy fractional pharmacokinetics model arising from drug assimilation into the blood stream. Abstract and Applied Analysis 2013, Article ID 304739 (2013)
30. Mazandarani, M., Vahidian Kamyad, A.: Modified fractional Euler method for solving Fuzzy Fractional Initial Value Problem. Commun. Nonlinear Sci. Numer. Simulat. 18, 12–21 (2013)
31. Mazandarani, M., Najariyan, M.: Type-2 Fuzzy Fractional Derivatives. Commun. Nonlinear Sci. Numer. Simulat. (2013), doi:10.1016/j.cnsns.2013.11.003
32. Wu, H.C.: The improper fuzzy Riemann integral and its numerical integration. Information Science 111, 109–137 (1999)
33. Dubios, D., Prade, H.: Towards fuzzy differential calculus-part3. Fuzzy Sets and Systems 8, 225–234 (1982)
34. Anastassiou, G.A.: Fuzzy Mathematics: Approximation Theory. STUDFUZZ, vol. 251. Springer, Heidelberg (2010)
35. Anastassiou, G.A., Gal, S.: On a fuzzy trigonometric approximation theorem of Weierstrass-type. J. Fuzzy Math. 9, 701–708 (2001)
36. Chalco-Cano, Y., Román-Flores, H.: On new solutions of fuzzy differential equations. Chaos, Solitons & Fractals 38, 112–119 (2008)
37. Canuto, C., Hussaini, M.Y., Quarteroni, A., Zang, T.A.: Spectral Methods in Fluid Dynamics. Springer, New York (1989)
38. Bonate, P.L.: Pharmacokinetic-pharmacodynamic modeling and simulation. Springer (2011)

A Multi-reference Ontology for Profiling Scholars' Background Knowledge

Bahram Amini, Roliana Ibrahim, Mohd Shahizan Othman, and Mohd Nazir Ahmad

Dept. of Information Systems, Faculty of Computing,
Universiti Teknologi Malaysia (UTM), Malaysia
avbahram2@live.utm.my, {roliana,shahizan,mnazir}@utm.my

Abstract. In most ontology-based scholar's recommender systems, profiling approaches employ a reference ontology as a backbone hierarchy to learn the topics of scholar's interests. It often works on the assumption that the reference ontology contains possible topics of scholars' preferences. However, such single reference ontologies lack sufficient ontological concepts and poor ontological concepts, which unable to capture the entire scholars' interests in terms of academic knowledge. In this paper, we extract, select, and merge heterogeneous subjects from different taxonomies on the Web and enrich by Wikipedia to constructs an OWL reference ontology for Computer Science domain. Compared to similar reference ontologies, our ontology purely supports the structure of scholars' knowledge, contains richer topics of the domain, and best fits for profiling the scholars' knowledge.

Keywords: Profiling, Ontology, Recommender System, Background Knowledge.

1 Introduction

The effectiveness of recommender systems depends highly on the completeness and accuracy of user profiles [1]. Much research had proven that ontology-based methods for profiling are effectively able to model user preferences and improve the recommendation accuracy [2]. It is also true for scholars' domain where a recommender system suggests articles to the researchers based on their background knowledge [3]. It profiles the knowledge that individual scholar captures through the research task. Most ontology-based profiling approaches such as [4] and [5] utilize a pre-built topic hierarchy which serves as a reference ontology. Such reference ontologies act as an initial model of scholars' preferences.

Many approaches utilize Open Directory Project (ODP) hierarchy as reference ontology [1]. However, the problem with such topic hierarchies is that the number of topics is insufficient, and the quality of the concepts is poor, leading to inaccurate scholars' profiles. Moreover, the concepts in scholar domain are growing over time and multidisciplinary. It motivates the use of live taxonomies which have been updated over time. Besides, to the best of our knowledge, there is no published

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_4, © Springer International Publishing Switzerland 2014

ontology on the Web which is able to represent the knowledge of scholars in Computer Science (CS) domain.

To address these issues, we construct a reference ontology by assembling multiple CS taxonomies, which supports sufficient topics of scholars' background knowledge. Precisely, we integrate several directories pertinent to CS domain, which fits to the profiling purpose and quality factors including completeness and richness.

The rest of the paper is organized as follows: Section 2 overviews the related work and motivates the research. We describe our framework for developing the reference ontology in Section 3. Section 4 represents an implementation of the framework. Section 5 deals with the evaluation. Section 6 discusses the results and suggests research extension. Finally, Section 7 draws conclusion.

2 Related Work

Many attempts have been conducted to develop user profiles by employing ontology-based methods. In most cases, the initial concepts are extracted from a general-purpose dictionary such as WordNet, ODP, or Yahoo Directory. The work in [6] engages ODP as a reference ontology for modeling users' preferences using the contextual information. ODP hierarchy is utilized to map the user's queries to the ontological concepts. It only employs three levels of the hierarchy with a minimum of five underlying documents. Sieg et al. [7] employ ODP as reference ontology and utilize associated Web pages as training data for representing the domain concepts. The textual information extracted from Web pages is represented by term vectors to specify the ontology concepts.

The work in [8] uses traditional cataloging scheme- the Library of Congress Classification (LCC), as reference ontology for document classification. The reference ontology is extracted from LCC, and the user's favorite terms are organized by using both borrowing records of individual user and notes keyed by librarian. Kodakateri et al. [9] summarize a subset of ACM Computing Classification System (CCS) concepts to create the user profile. ACM taxonomy has been used as reference ontology to categorize both visited and unvisited documents of CiteSeerX digital library. Hence, the scholar's profile is a list of ACM concepts and their corresponding weights, ordered based on the weights.

However, existing ontologies lack sufficient concepts, lack domain-specific concepts, and provide low domain coverage. These shortcomings hamper the exploitation of such ontologies as reference ontology for modeling scholars' knowledge. For example, about 85% of ODP subjects are useless for profiling CS's scholars' knowledge. In addition, they do not support "part-of" relation among the topics.

3 Methodology

Our method involves two parts: In the first part, we describe the characteristics of several source taxonomies, topic selection from resources, and merging approach to develop a primary artifact. In the second part, we enrich the primary artifact with the Wikipedia topics and reorganize the topics' relationships using Wiki pages.

3.1 The Framework

To construct the reference ontology, we exploit the following six free Web taxonomies, which provide topical information for CS domain:

1. Open Directory Project (ODP) (www.dmoz.org)
2. Best of the Web (BOTW) (www.botw.org) [10]
3. JoeAnt Directory (www.joeant.com/)
4. World Wide Web Virtual Library (VLIB) (http://vlib.org/Computing)
5. ACM/IEEE Curriculum 2008 (www.acm.org/education) [11]
6. Wikipedia (www.wikipedi.org)

Wikipedia plays two important roles: as a source taxonomy and as an enrichment tool for reference ontology. Fig. 1 represents the structure of topic selection, composition, and enrichment. Since each taxonomies provides different hierarchy in different granularity, the process of concept selection and merging is applied [12]. Moreover, formal education is an important part of scholars' knowledge since scholars gain academic knowledge through the formal education by taking courses at higher educations. Therefore, concepts representing scholars' formal educations are extracted from a standard curriculum, ACM/IEEE Computing Curriculum 2008 [11] that is a reference subject hierarchy which provides a full course based and community consensus terminologies of body of scholars' knowledge [2]. This curriculum offers full content and a hierarchical arrangement of CS topics.

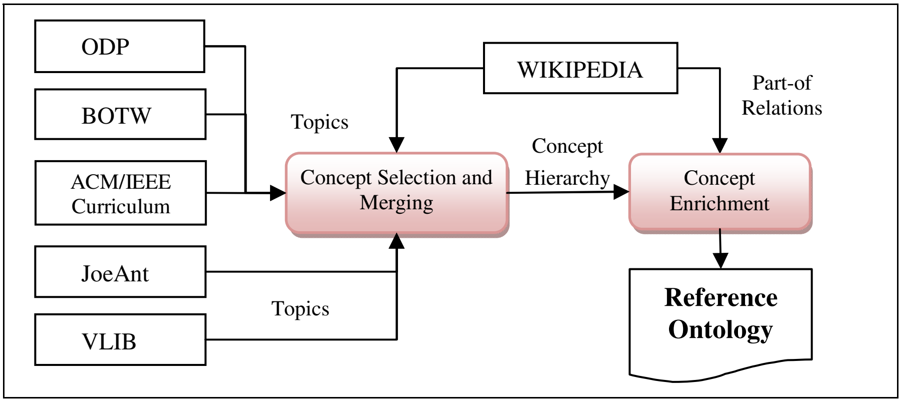


Fig. 1. The framework of ontology development for Computer Science domain

3.2 Characteristics of Source Taxonomies

We recognized two types of heterogeneity among the source taxonomies which negatively influence the integration process including schematic (naming) and structural conflicts. The structural conflict refers to the differences in the organization of information (subjects) while the schematic conflict deals with differences in syntax. Table 1 represents a sample of heterogeneity between the entries of ODP and Wikipedia. As shown, the topic "Particle_Swarm" in ODP and Wikipedia is represented in different syntax. The organization of the data is also different: it is a sub topic of two different supper topics:

1- Computer/Artificial_Life,
2- Artificial Intelligence/Machine Learning/Evolutionary Algorithms.

To deal with such heterogeneities, the syntactic and semantic similarities among the topics are verified and aligned into a unified form by means of Wikipedia taxonomy as a domain consensus knowledge resource. It is a rich source of knowledge about the latest topics of CS, and an expert agreement encyclopedia [13], which provides "part-of" relation of CS topics.

Table 1. A view of two source taxonomies representing the structural and naming heterogeneity among the ODP and Wikipedia entries

ODP	Wikipedia
Artificial_Life/**Particle_Swarm**	1.0.0 Artificial Intelligence
Artificial_Life/Particle_Swarm/Conferences	1.1.0 Machine Learning
Artificial_Life/Particle_Swarm/Papers	1.1.4 Evolutionary
Artificial_Life/Particle_Swarm/People	1.1.5 Algorithms
Artificial_Life/People	Genetic Algorithms
Artificial_Life/Publications	Ant Colony Optimization
Artificial_Life/Publications/Journals	**Particle Swarm Optimization**
Artificial_Life/Publications/Papers	Bees Algorithm
Artificial_Life/Research_Groups	
Artificial_Life/Software	

3.3 Merging Method

We use ODP as "working hierarchy" and integrate the other taxonomies with it step by step. The schematic and structural heterogeneities among topics are resolved by two methods: 1) matching the topics/concepts using the syntactic similarities, 2) aligning the concepts in terms of relationship by focusing on Wikipedia taxonomy. To do this, an extension of SMART algorithm [14] which uses Wikipedia as conflict resolver is employed. Below is the algorithm which exploits syntactic as well as semantic similarity methods for comparing respected topics:

1. Input: Working Ontology (WO) and Augmenting Taxonomy (AT).
2. Make two lists of topics from WO and AT

3. For each AT's topic scan both lists to find similar topics:
    a. For syntax similarity: look for synonymy, common substrings, shared prefixes, or shared suffixes.
    b. For semantic similarity: engage Wikipedia to measure the topic relatedness using the corresponding Wiki pages.
4. For each pair of topics do: if <similar> then ignore AT's topic, else find a parent using Wikipedia taxonomy and add AT's topic to WO under that parent topic.
5. Repeat steps 3-5 until no more AT's topic is available.

4 Implementation

4.1 Concept Selection and Merging

In order to control the vocabulary of reference ontology, we first identified the core topics of CS domain from the topics of ten international conferences held between 2009 and 2013. We found out that the topics are spanned in a wide spectrum, ranging from theoretical algorithms to practical issues of computing in hardware. Thus, we categorized the topics to four main areas and nine second-level areas as follows:

- Theory of computation
- Algorithms and data structures
- Programming methodology and languages
- Computer elements and architecture

* Numerical and symbolic computation
* Computer-human interaction
* Computer networking and communication
* Parallel and distributed computation
* Artificial intelligence
* Operating systems
* Database systems
* Computer graphics
* Software engineering

Next, ODP has been used as a primary hierarchy and all relevant CS topics using the above controlling topics are extracted. Currently, ODP contains 786,225 subjects, spanning from 3 to 5 levels. The CS segment contains 8,471 entries. Thus, we applied domain consensus filters iteratively in five stages to sift irrelevant topics, i.e., filtering the topics that appear frequently in CS context but do not appertain to the scholar's domain. In other words, ODP contains a great number of topics that are not a research topic such as Groups, FAQ and Help, Journals, Clip Arts, Advertising, Companies, etc.

Table 2 depicts the filter description, typical examples, and the number of resulting entries for each filtering process. As shown, engaging the filters assists to decrease the total number of concepts to 747 subjects (about %8 of the original items) and makes the final topics more relevant to the scholars' knowledge.

Table 2. Five types of filters for cleaning the ODP topics to CS topics

Filter	Irrelevant Subjects	Examples	Final topics
1	Hardware components, Specific Software, Education	Hard Disk, Peripheral, Storages, Drivers, Home Automation, Personal Tools, Freeware, Organizations, Training Resources.	2,303
2	Internet Services, Commercial Tools, Organizations	Chat, Groups, Messengers, Broadcasting Tools, Domain names, Weblogs, Web Hosting, Wikis, Trademark, Tools, Brands, Markets, Mailing Lists, ERP, Service providers, Software Foundations	1925
3	Projects, Programming, Information Technologies	Open Source, Platforms, Compilers, Methodologies, Agents, Networking, Engineering, etc.	1327
4	Applications, Products	Programming Languages, Tools, Development Kits, Technologies, Libraries, Operating System, Compilers, DBMS, Cookies, Plug-in, Protocols	998
5	Non CS topics	Directories, Ethics, History, Help, QA, Country Name, Management, Configuration, Administration, etc.	747

Finally, the topics and their relationships from BOTW, JoeAnt, VLIB taxonomies are incrementally augmented to the primary hierarchy. In this stage, cross-checking of topics and inclusion of missed topics for three topmost levels are performed. In practice, we made an OWL version of each ontology for each taxonomy using Protégé 4.2 [15] and then combined them as a whole structure by SWOOP framework [16]. The later framework also controls the inconsistencies, duplicates, and structural properties of the ontology. In addition, the heterogeneities among ontological concepts are resolved. The resulting hierarchy at this stage contained nine top concepts (classes), 242 disjoint classes, and 4.2 classes per level on average. The right side of Fig. 2 represents a small part of the ontology in Protégé. As shown, the arrangement of topics looks like a hierarchy but requires more refinements.

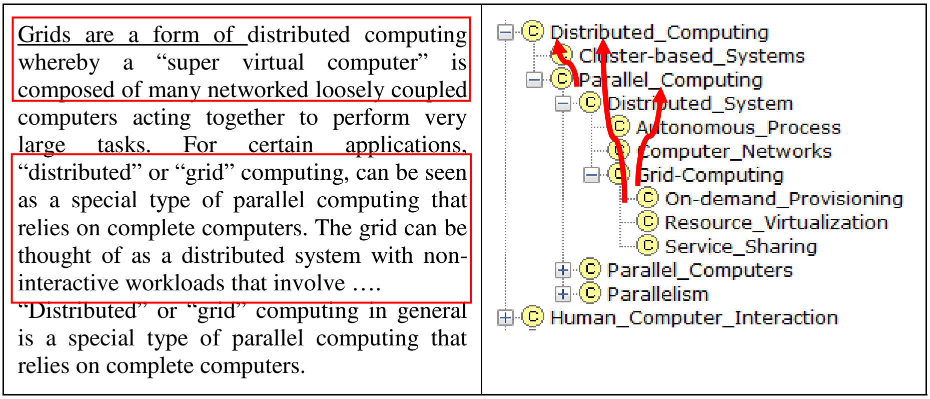
4.2 Concept Enrichment by Wikipedia

To enrich the hierarchy with more grained topics, we employed Wikipedia encyclopedia. It provides "part-of" relation while other taxonomies lack of sufficient information about the topics' relations [17]. Each Wiki article contains full description of CS topics and the respected subtopics which describe the synonym subjects. Fortunately, Wikipedia provides lower/upper level topics for CS domain: Wiki articles have been assigned to categories, and the categories have been assigned to more general categories, making a hierarchy of topics.

We exploited such relations to incorporate Wikipedia's topics as well as reorganize the topics in the developing hierarchy. For example, there is no subtopic for "Grid

Computing" topic in the other taxonomies; therefore, we located a topic in Wikipedia that linked to it, either as a child or parent. Fig. 2 represents how a textual knowledge of Wikipedia is utilized to link "Grid Computing" and "Distributed System" topics to their upper level topic "Parallel Computing".

We extracted the textual information from Wikipedia articles which contained an occurrence of favorite topics, i.e., the incomplete topics that had zero or a few child. This task both enhanced the topics in the hierarchy at the same level or incorporated new topics to the developing hierarchy. However, in the case of multiple Wiki pages found for a particular topic, the semantic relatedness [18] is used to disambiguate the offspring or parent topics. Neither tedious nor laborious effort is required to transform the Wiki page's content to adapt to the hierarchy. Finally, orphan topics such as "Proactive learning" that has no link to the Wikipedia articles are excluded.



Grids are a form of distributed computing whereby a "super virtual computer" is composed of many networked loosely coupled computers acting together to perform very large tasks. For certain applications, "distributed" or "grid" computing, can be seen as a special type of parallel computing that relies on complete computers. The grid can be thought of as a distributed system with non-interactive workloads that involve ….
"Distributed" or "grid" computing in general is a special type of parallel computing that relies on complete computers.

Fig. 2. Textual knowledge of Wiki pages are used to enrich the working reference ontology

4.3 Concepts Elicitation from Formal Education

To augment scholars' formal education to the reference ontology, the learning subjects in ACM/IEEE curriculum is applied. It contains 14 major areas, 161 units, and 1152 cores as well as their concepts. Fig. 3 represents a partial view of the curriculum representing the "Intelligent Systems" area of Information Systems (IS), and "Machine Learning" units, the cores and the corresponding concepts. Here, the cores and concepts are the target topics which we collected from the curriculum through the following process:

Firstly, an intuitive filter is designed to perform the initial processing. This filter excludes the undergraduate cores as well as very general subjects/topics such as "Definitions", "Applications", "Programming Skills", "Introductions", and "Hardware", which are not pertinent to research domain of CS domain. The "controlling topics" in Section 4.1 were an appropriate guideline to filter out such general topics. For instance, the topics "Programming Fundamentals", "GUI Programming", "Networked Applications", and "Data Types" from the unit and core sections are pruned because met the filters criteria. The remaining areas and cores, that are %54 of the total, are merged to the reference ontology. Moreover, corresponding concepts, which contain a list of fine-grained and much richer ontological concepts are included using "Label Annotation" in Protégé.

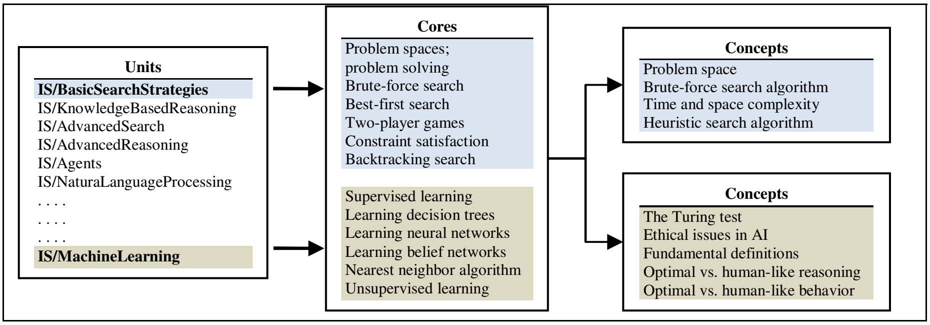


Fig. 3. A subject hierarchy collected from ACM/IEEE Curriculum 2008 [11]

5 Evaluation

Ontology evaluation is much important when it is constructed from different heterogeneous knowledge resources, leading to conflict instances, and decreasing the usefulness of the ontology [19]. In the literature, there are many approaches for evaluating the ontologies which deal with different functional and structural aspects of the ontology [20]. However, depending on the case, some metrics must be adopted. Thus, we selected two metrics which are applicable and fit most in our problem:

1. Comparing with golden ontologies
2. Structural validation

- **Comparing with Golden Ontologies**

The goal is to determine the ontology richness compared to some base ontologies. Ontology richness is measured based on the amount of relational information spreading over the ontology compared to the average number of topics/classes. The relations is measured by total classes, the number of classes per levels, and average siblings (or branching factors, BF) of classes. It is inspired by the question of how the conceptual knowledge is distributed over the ontology. Therefore, a measurement of richness is provided by comparing the properties with the base ontologies [21].

Our reference ontology consists of 15 top levels and 2035 concepts in total. It also has 7 maximum levels, 2 minimum levels, and 4.4 levels in average. The maximum branching factor is 29, and BF per class is 5.6. Moreover, the capability of using annotations for fine-grained concepts enables the multi-language concept modeling, where knowledge items from different languages can be assigned to the concepts, providing flexible ontology without increasing the number of concepts and levels.

We also compare our reference ontology with six related works (Section 2). Table 4 outlines the distribution of classes and the relationships across different ontologies. As described in Section 2, the ontological concepts are too general and pertaining to low level terms of the domain. Our reference ontology contains a higher number of top classes (15), higher number of average levels (deepness) compared to the number of classes (4.4), and a higher average branching factor compared to the levels (5.6). Additionally, the total number of classes is 2,035, which is an improvement for hierarchical reference ontology in CS domain.

- **Structural Validation**

The main goal of this measure is to avoid using inconsistent or incorrect reference ontology for profiling. The structural validation aims to check the ontology against the consistency and correctness. It is a type of diagnostic task over ontology properties and attributes. To perform such evaluations, SWOOP[1] tool was employed for automatic detection of possible inconsistency in the ontology [22]. The SWOOP's reasoner reliably detects and assists in fixing the structural errors. Having engaged SWOOP, we first performed automatic diagnosing test over the ontology to understand the cause and source of problematic concepts. Then, caring about the cost and benefits, some amendments was applied. The final checking showed that no syntactical problems such as cycles, disjoint partitions, or redundant concepts longer exist.

Table 3. The comparison of different reference ontologies in terms of ontology richness

Approaches	Type of Ontology	Total Classes	Top Classes	Average Levels	Average BF
Trajkova et al. [23]	ODP	2,991	13	3.0	6.1
Sieg et al. [24]	ODP	4,470	11	3.0	7.4
Sieg et al. [25]	ODP	563	11	6.0	4.0
Mohammed et al. [6]	ODP	NA	11	3.0	NA
Liao et al. [8]	LCC	853	13	2.0	3.5
Kodakateri et al. [9]	ACM CCS	1470	11	3.2	5.0
Our Approach	**Merged**	**2,035**	**15**	**4.4**	**5.6**

6 Discussion and Research Extension

In this paper, we started with the issues of engaging single reference ontologies in modeling scholars' background knowledge. We presented an approach which integrates multiple heterogeneous taxonomies of CS domain. The main contribution of our approach lies in integrating several taxonomies including ODP, Wikipedia, and others to develop a suited reference ontology for profiling the scholars' knowledge. Our approach is flexible since we could incorporate several taxonomies to enrich the

[1] `http://code.google.com/p/swoop/`

ontology over time. Our approach firstly exploited domain ontologies which provided relatively basic concepts of the domain, and secondly, employed Wikipedia which assists in concept enrichment and efficient merging process. Additionally, our reference ontology, compared to the base ontologies, contained richer ontological concepts.

In addition, our approach is independent to the number of taxonomy resources. This flexibility inspires the incorporation of new hierarchical ontologies to the framework, which consequently improves the ontology richness. However, the degree of heterogeneity among the hierarchies is a challenging issue that should be resolved in the future. As more hierarchies are included, the complexity, cost of merging, and alignment of various concepts will be increased.

7 Conclusion

In this paper, we merged multiple taxonomies of Computer Science domain and developed a richer reference ontology which can be used for modeling scholars' background knowledge. Wikipedia as a shared expert knowledge enabled us to augment and enrich the elements of ontological concepts derived from other taxonomies. To validate the reference ontology, no external contribution such as expert assessment was employed that made our approach flexible and domain independent. Moreover, this study made non-obvious connections between different taxonomies in the domain by semantically matching concepts by Wikipedia. Furthermore, the use of ODP, BOTW, JoeAnt, VLIB, and relevant taxonomies enriched the domain concepts and facilitated the domain cross-validation. We also considered a broader range of ontological concepts compared to the existing ones that provided the reference ontology with deeper areas of scholars' knowledge, assisting in improved scholars' profiling. Besides, we provided more levels in the hierarchy, which enabled more scholar knowledge representation as well as preserving core areas of scholars' knowledge.

Acknowledgment. This work is supported by the Ministry of Higher Education (MOHE) and Research Management Centre (RMC) at the Universiti Teknologi Malaysia (UTM) under Exploratory Research Grant Scheme (Vote No. R.J130000.7828.4L051).

References

1. Gauch, S., Speretta, M., Chandramouli, A., Micarelli, A.: User Profiles for Personalized Information Access. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) Adaptive Web 2007. LNCS, vol. 4321, pp. 54–89. Springer, Heidelberg (2007)
2. Liao, I., Hsu, W., Chen, M.-S., Chen, L.: A library recommender system based on a personal ontology model and collaborative filtering technique for English collections. Emeral Gr. Publ. 28(3), 386–400 (2010)

3. Amini, B., Ibrahim, R., Othman, M.S., Rastegari, H.: Incorporating Scholar's Background Knowledge into Recommender System for Digital Libraries. In: 5th Malaysian Conference in Software Engineering (MySEC), pp. 516–523 (2011)
4. Middleton, S.E., De Roure, D., Shadbolt, N.R.: Ontology-Based Recommender Systems. In: Handbook on Ontologies, International Handbooks on Information Systems, pp. 779–796 (2009)
5. Schiaffino, S., Amandi, A.: Intelligent User Profiling. In: Bramer, M. (ed.) Artificial Intelligence An International Perspective. LNCS (LNAI), vol. 5640, pp. 193–216. Springer, Heidelberg (2009)
6. Mohammed, N., Duong, T.H., Jo, G.S.: Contextual Information Search Based on Ontological User Profile. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ICCCI 2010, Part II. LNCS, vol. 6422, pp. 490–500. Springer, Heidelberg (2010)
7. Sieg, A., Mobasher, B., Burke, R.: Ontological User Profiles for Personalized Web Search. In: AAAI Work. Intell. Tech. Web Pers., pp. 84–91 (2007)
8. Liao, S., Kao, K., Liao, I., Chen, H.: PORE: a personal ontology recommender system for digital libraries. Library (Lond) 27(3), 496–508 (2009)
9. Kodakateri, A., Gauch, S., Luong, H., Eno, J.: Conceptual Recommender System for CiteSeerX. In: RecSys 2009, pp. 241–244 (2009)
10. Mahan, R.: Best of the Web. Sci. Am. Mind 19, 84–85 (2008)
11. Seidman, S., McGettrick, A.: Computer Science Curriculum 2008: An Interim Revision of CS 2001 Report from the Interim Review Task Force (2008)
12. Gal, A., Shvaiko, P.: Advances in Ontology Matching. In: Dillon, T.S., Chang, E., Meersman, R., Sycara, K. (eds.) Advances in Web Semantics I. LNCS, vol. 4891, pp. 176–198. Springer, Heidelberg (2009)
13. Medelyan, O., Milne, D., Legg, C., Witten, I.H.: Mining meaning from Wikipedia. J. Hum. Comput. Stud. 67(9), 716–754 (2009)
14. Noy, N.F., Musen, M.A.: An Algorithm for Merging and Aligning Ontologies: Automation and Tool Support. In: Workshop on Ontology Management at the 16th National Conference on Artificial Intelligence (AAAI 1999), pp. 17–27 (1999)
15. Horridge, M., Jupp, S., Moulton, G., Rector, A., Stevens, R., Wroe, C.: A Practical Guide To Building OWL Ontologies Using Protege 4 and CO-ODE Tools, pp. 11–102 (2007)
16. Kalyanpur, A., Parsia, B., Sirin, E., Grau, B.C., Hendler, J.: Swoop: A Web Ontology Editing Browser. Web Semant. Sci. Serv. Agents World Wide Web 4(2), 144–153 (2006)
17. Medelyan, O., Milne, D.: Augmenting Domain-Specific Thesauri with Knowledge from Wikipedia. In: Proc. NZ Comput. Sci. Res. Student Conf. NZ CSRSC 2008 (2008)
18. Gabrilovich, E., Markovitch, S.: Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence, pp. 6–12 (2007)
19. Tartir, S., Arpinar, I.B., Sheth, A.P.: Ontological Evaluation and Validation. In: Poli, R. (ed.) Theory and Applications of Ontology: Computer Applications, pp. 115–130. Springer Science+Business Media (2010)
20. Brank, J., Grobelnik, M., Mladenić, D.: A Survey of Ontology Evaluation Techniques. In: Proceedings of the Conference on Data Mining and Data Warehouses (SiKDD 2005) (2005)
21. d'Aquin, M., Schlicht, A., Stuckenschmidt, H., Sabou, M.: Criteria and Evaluation for Ontology Modularization Techniques. In: Stuckenschmidt, H., Parent, C., Spaccapietra, S. (eds.) Modular Ontologies. LNCS, vol. 5445, pp. 67–89. Springer, Heidelberg (2009)

22. Obrst, L., Ashpole, B., Ceusters, W., Mani, I., Smith, B.: The Evaluation of Ontologies: Toward Improved Semantic Interoperability. In: Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences, pp. 1–19 (2007)
23. Trajkova, J., Gauch, S.: Improving Ontology-Based User Profiles. In: Recherche d'Information Assistee par Ordinateur (RIAO 2004), pp. 380–390 (2004)
24. Sieg, A., Burke, R.: Web Search Personalization with Ontological User Profiles. In: CIKM 2007, pp. 525–534 (2007)
25. Sieg, A., Mobasher, B., Burke, R.: Learning Ontology-Based User Profiles: A Semantic Approach to Personalized Web Search. IEEE Intell. Informatics Bull. 8(1), 7–18 (2007)

A New Binary Particle Swarm Optimization for Feature Subset Selection with Support Vector Machine

Amir Rajabi Behjat[1], Aida Mustapha[1], Hossein Nezamabadi-Pour[2],
Md. Nasir Sulaiman[1], and Norwati Mustapha[1]

[1] Faculty of Computer Science and Information Technology
University Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia
rajabi.amir6@gmail.com,
{aida_m,nasir,norwati}@upm.edu.my

[2] Department of Electrical Engineering, Shahid Bahonar University of Kerman
P.O. Box 76169-133, Kerman, Iran
nezam@mail.uk.ac.ir

Abstract. Social Engineering (SE) has emerged as one of the most familiar problem concerning organizational security and computer users. At present, the performance deterioration of phishing and spam detection systems are attributed to high feature dimensionality as well as the computational cost during feature selection. This consequently reduces the classification accuracy or detection rate and increases the False Positive Rate (FPR). This research is set to introduce a novel feature selection method called the New Binary Particle Swarm Optimization (NBPSO) to choose a set of optimal features in spam and phishing emails. The proposed feature selection method was tested in a classification experiments using the Support Vector Machine (SVM) to classify emails according to the various features as input. The results obtained by experimenting on two phishing and spam emails showed a reasonable performance to the phishing detection system.

Keywords: Particle swarm optimization, feature selection, phishing, spam, social engineering, SVM.

1 Introduction

Many IDSs are using database of well-known actions to compare normal and abnormal data or activities for sending alerts when a match is detected [1], [2]. Attackers evade Intrusion Detection Systems (IDS) using various ways, such as using the old unknown attack, hiding an attack in a concealed or encrypted channel, as well as posing as social engineering attacks [3]. Social engineering (SE) is a developing science that capitalizes on human trusty nature and is a serious threat to all organizations [4], [5]. Most familiar social engineering attacks include phishing and spam emails that convince users to open emails with abnormal links, pictures, videos, and even URLs [3], [6]. Phishing, spam, and

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_5, © Springer International Publishing Switzerland 2014

even legitimate emails are basically similar in the style and content. Nonetheless, beyond the content, the structural and other special features will be able to make a distinction whether the emails are phishing emails or spam emails. Phishing is considered as a subcategory of spam [7].

In detecting phishing and spam/legitimate emails, the feature selection quality along with computational methods are required to guarantee the effectiveness of a classification/detection system [3]. This means the elimination of irrelevant features via the feature reduction process will increase the accuracy and reduce the false positive rate during detection since a smaller number of feature sets up the speed of the computation [8], [9]. Most studies have considered various features of phishing and spam emails [1], [10], [11], [12], [13]. The accuracy of phishing and spam detection showed a good result in a number of studies [12], [14], [15] while the number of features multiplies the computational cost and decreases the accuracy [16]. Generally, the lack of knowledge related to the false positive and the impact of features on the accuracy will reduce the performance of phishing detection [14].

The main objective of this study is to select a combination of features in phishing emails and evaluate the impact of these features on the basis of computational cost, false positive rate, and accuracy percentage in detecting phishing emails. This study will propose a New Binary Particle Swarm Optimization (NBPSO) for feature selection and will test the performance via a classification experiment with an existing Support Vector Machine (SVM) classifier. This study attempts to prove that the high classification accuracy and low false positive rate are possible through feature reduction that should result in lower dimensionality in feature sets, which covers important parts of emails such as subjects, bodies, links, URLs and attached files within the email body.

The paper organization keeps on as follows. Section 2 begins with the principle of Support Vector Machine (SVM). Section 3 introduces the principles of Particle Swarm Optimization (PSO) and Binary PSO preparation. Section 4 details out the experimental results, Section 5 discusses the ROC curve and AUC analysis, and finally Section 6 concludes the work and sets future research.

2 Principles of Support Vector Machine (SVM)

In 1995, Guyon, Boser, and Vapnik [17] introduced the Support Vector Machine (SVM). SVM is based on statistical learning theory and is able to prevents overfitting in classification, hence is well-known for its high classification accuracy. The SVM classifier predicts a new instance into a predefined category based on given training examples as shown in Equation 1:

$$D = (o_i, y_i) | o_i \in R_p, y_i \in \{-1, 1\}_{i=1}^{pt} \tag{1}$$

where pt is the number of samples and (o_i, y_i) shows the i^{th} training sample with its corresponding labels. $o_i = (o_i, o_{i_2}, o_{i_3}, \ldots, o_{i_p})$ is a p-dimensional vector in the feature space as shown in Equation 2.

$$\min 1/2\langle w \cdot w\rangle + C\sum_{i=1}^{pt} \zeta_i, y_i(\langle w \cdot o_i\rangle + b)\zeta_i \geq 0 + \zeta_i - 1 \geq 0 \tag{2}$$

where C is the penalty parameter that controls the decision function complexity and the number of misclassified training examples. ζ_i is the positive slack variable. The hyperplane which has the largest distance to the nearest training data point will create a suitable separation. This model can be solved using the introduction of the Langrage multipliers $0 \leq \alpha_i \leq C$ for dual optimization model [18], [19], [20]. The classifier function and the optimal b^* and w^* can be defined after achievement of the optimal solution α_i based on Equation 3.

$$\text{sign}\langle w^* \cdot o_i + b^*\rangle \text{ or } \text{sign}(\sum_{i=1}^{pt} y\alpha_i^* \langle o_i \cdot o\rangle + b^*)oi.o + b*) \tag{3}$$

The SVM maps training data nonlinearly within a high-dimensional feature space by kernel function $k(o_i, o_j)$ where linear separation may be possible. The kernels will decrease a complex classification task by separating hyperplanes. The typical kernel function given as in Equation 4.

$$k(o_i, o_j) = \exp(\frac{-1}{\delta^2}\|o_i - o_j\|^2) = \exp(-\gamma\|o_i - o_j\|^2) \tag{4}$$

The SVM classifier then changes to the following model after choosing the kernel function shown in Equation 5.

$$\text{sign}(\sum_{i=1}^{pt} \gamma_i \alpha_i^* \langle o_i \cdot o\rangle + b^*) \tag{5}$$

The performance of an SVM classifier is highly dependent on C and γ, which are the hyper-parameters. These two parameters will affect the number of support vectors and the size of margin in the SVM [20].

3 Principles of Particle Swarm Optimization (PSO) and Binary PSO Preparation

Particle Swarm Optimization (PSO) was introduced in 1995 based on the behavior of swarming animals. This algorithm has been used for optimization in different fields such as data clustering, optimization of artificial neural network, and network wireless [15], [16]. The set of particles builds a population (swarm) of candidate solutions. PSO is similar to heuristic algorithms in the sense that it searches some solutions within the initialized population. However, unlike Genetic Algorithm (GA), PSO does not follow operators such as mutation and crossover [7], [16].

In PSO algorithm, each particle is a point in D-dimensional space, so the i^{th} particle is represented as $X_i = (x_{i_1}, x_{i_2}, \ldots, x_{i_s})$. Because PSO calculates the best fitness rate (*pbest*) according to previous position of each particle, the rate for any particle is $P_i = (p_{i_1}, p_{i_2}, \ldots, p_{i_s})$. The global best and velocity of particle i are '*gbest*' and $V_i = (v_{i_1}, v_{i_2}, \ldots, v_{i_s})$, respectively. Meanwhile, the manipulation of each particle is continued as the following Equation 6 and Equation 7.

$$v_{id} = w * v_{id} + c1 * rand() * (p_{ad} - x_{id}) + c2 * Rand() * (p_{ad} - x_{id}) \tag{6}$$

$$x_{id} = x_{id} + v_{id} \tag{7}$$

where w is the inertia weight, c_1 and c_2 are the stochastic acceleration weighting that leads particles toward *pbest* and *gbest* positions. $rand()$ and $Rand()$ are the random functions between [0,1]. $Vmax$ shows the velocity of each particle.

The New Binary Particle Swarm Optimization (NBPSO) algorithm follows the action of chromosomes in GA, so it is coded such as a binary string. In the specific dimension, the particle velocity is used like a probability distribution with the main role to randomly produce the particle position. Updating the particle position follows Equation 8, whereby the sigmoid function is used to identify new particle position based on binary values.

$$S(v_{id}) = Sigmoid(v_{id}) = \frac{1}{1 + e^{-v_{id}}} \tag{8}$$

$$\text{If } rand < S(v_{id}(t+1)) \text{ then } x_{id}(t+1) = 1 \text{ else } x_{id}(t+1) = 0 \tag{9}$$

where $rand$ is a random value between [0, 1] and v_{id} is limited to $vmax()$. In each dimension, a bit value 1 shows the selected feature may participate for the next generation. On the other hand, a bit value of 0 is not required as a relevant for next generation [8].

3.1 Drawbacks of Current PSO

The particles in continuous Particle Swarm Optimization (PSO) algorithm are defined by x and v values. The particle at position (x) is a potential solution and v is the speed of each particle that shows the future position of a particular particle relative to its current position. Large value of v shows that the particle position is not suitable, hence the value should change towards an optimal solution. The small value of v demonstrates that the particle position is moving towards optimal solution or with 0 value.

There are different definitions of x and v in binary particle swarm optimization algorithm (BPSO). The speed of particle (v) in this algorithm shows 0 or 1 for the position of particle (x) instead of finding optimal solutions. In other word, the v_{id} identifies the x_{id} value (0 or 1). Since the probability of x_{id} should be between 0 and 1, then v_{id} uses the sigmoid function as previously shown in Equation 8.

In BPSO, the large value of x_{id} (towards positive values) meaning x_{id} is near to 1 and the small value (towards negative values) reduces the probability of

1 for x_{id}. On the other hand, if v_{id} is 0, then the value of x_{id} will be changed to 0 or 1 with the probability of 50%. In addition, the value of x_{id} is identified regardless the previous value or position. Based on these scenario, there are two drawbacks in the BPSO as algorithm deliberated as follows.

The first drawback lies in the sigmoid function. Conceptually, the large value of v_{id} towards negative or positive values shows that x_{id} position should change for a specific dimension. However, in the binary particle swarm optimization, v_{id} steers x_{id} towards 0 or 1. Additionally, the speed of particle (v) near to 0 shows that the position of particle (x) is satisfied and the sigmoid function demonstrates an equal probability of 0 or 1 for x_{id}.

The second drawback is the process to update particle position (x). In the average of initial iterations, all the particles come up the optimal solution. Nonetheless, these particles keep out the optimal solution even after several iterations. This means the optimal solution may be near to 0, but the probability of 0 or 1 decrease to 50% during such times.

3.2 Proposed New Binary Particle Swarm Optimization

Both drawbacks in the Binary Particle Swarm Optimization (BPSO) algorithm may be resolved using suitable functions and by updating the particle position (x_{id}) as shown in Equation 11. In this algorithm, the sigmoid function is replaced to $S'(v_{id})$ as shown in Equation 10.

$S'(v_{id})$ proves that the value of v_{id} towards positive values is the same as the negative values. Whenever the speed of particle (v_{id}) is near to 0 value, the output of function increases and moves to 0 too. On the other hand, for updating particle position in Equation 6 is replaced to the one in Equation 3. Finally, the large v_{id} value demonstrates that the particle position is not suitable and changes towards 0 or 1 while the small value of v_{id} decreases the probability of the changes in the position of particle (x_id). On the other hand, the 0 value of v_{id} will fix the particle position.

$$S'(v_{id}) = |tanh(\alpha x)| \tag{10}$$

$$\begin{aligned} &\text{If } rand < s'(v_{id}(t+1)) \text{ then } x_{id}(t+1) = complement(x_{id}) \\ &\text{else } x_{id}(t+1) = x_{id}(t) \end{aligned} \tag{11}$$

In this study, NBPSO finds an optimal binary vector, where each bit is associated with a feature. If the i^{th} bit of this vector equals to 1, the i^{th} feature will be allowed to participate in the classification. If the bit is a zero (0), the feature cannot participate in the classification process. Each resulting subset of features will be evaluated according to its classification accuracy on a set of testing data in an SVM classifier. We will divide the entire features by their importance and eliminate irrelevant features, which is indicated by the lowest ranked during the process. In other words, we will select important features by using the variable of the importance value that is based on their repetition in two classes. This strategy enables our approach to reduce the computational expenses of the dataset as well as to enhance the detection rates and reduce the feature dimensionality.

4 Experiments and Results

This study evaluated the classification accuracy or detection rate of New Binary Particle Swarm Optimization (NBPSO) algorithm for feature selection. The classification experiment used the Support Vector Machine (SVM) trained with the measurement vectors of 14,580 spam, ham, and phishing emails. A total of 1,620 measurements were available for testing. The experiments were performed using the Intel Pentium IV processor with 2.7GHz CPU, 4GB RAM, and Windows 7 Operating System with MATHWORK_R2010b development environment. The classification experiments used three well-known datasets in shown in Table 1.

Table 1. Dataset and the number of class

No.	Dataset	Size
1	SpamAssassin	6,954
2	SpamEmail	1,895
3	PhishingCorpus	4,563

In order to select a set of combined features in within the pool of phishing email, we applied the NBPSO to choose the best features within the extracted features as reported in the previous studies [1], [10], [11], [12], [13]. The results showed that NBPSO was able to search the complex space with a big number of features. Furthermore, NBPSO found an optimal binary vector, where each bit was associated with a feature. If the i^{th} bit of this vector equals to 1, the i^{th} feature will be allowed to participate in the classification process; but if the bit equals to 0 (zero), the feature cannot participate in the classification process. Each resulting subset of features was evaluated according to its classification accuracy on a set of testing data using the SVM classifier.

In order to evaluate the selected features based NBPSO as the feature selection method, we divided the features in each category and combined the categories in four classifiers such as 2C, 3C, 4C and 5C, so category 2, 3, 4, and 5 were divided into each classifier respectively. This analysis identified the best combination and the created detection rate by them. The best classifiers based on different categories with selected relevant features are related to 3C = C1, C4, and C5, 4C = C1, C2, C4, and C5, 5C = C1, C2, C3, C4, and C5 and 2C = C1 and C5 classifiers with respective detection rate as shown in Table 2. The best false positive rate (FPR) achieved was for 4C with 0.1%.

While previous studies either reported the FPR or complement the results with accuracy rate, we believe that we could improve these two rates near to 100 accuracy and and 0 for FPR, respectively. On the other hand, the number of

Table 2. The detection rate for each combination of features

Features	Feature Combinations	Detection Rate (%)
2C	C1,C5	91.49
3C	C1,C4,C5	98.99
4C	C1, C2,C4,C5	97.77
5C	C1,C2,C3,C4,C5	94.22

Table 3. CPU time and elapsed time to select the best combination of features

Features	Feature Combinations	Elapsed Time (s)	CPU Time (%)
2C	C1,C5	152.23	28
3C	C1,C4,C5	173.56	37.2
4C	C1, C2,C4,C5	198.38	46.45
5C	C1,C2,C3,C4,C5	215.67	63.65

features in each category influenced the performance and computational cost of the classifier. For example, 5C with high number of features has a higher CPU time and computational cost, which is 63% and elapsed time to 215.67(s) during the feature selection process as shown in Table 3.

For each dataset, the experiments were repeated in 10 runs based on 4 tests (refer to Figure 1). The detection rate is based on classification accuracy, hence the parameters for NBPSO algorithm are set as $\alpha = 1$. The population size is set to 20, C1 and C2 are set to 2 and the weight values lie between 0.5 to 1.5. Note that the SVM classifier was trained and tested by 90:10 percentage split in each dataset respectively. The obtained classification accuracy was illustrated by the form of 'average ± standard deviation'. The results showed that the proposed NBPSO-based feature selection with the SVM classifier resulted in higher detection rate across all datasets as shown in Table 2.

Figure 1 illustrates our evaluation based on 4 tests, which means in the first and the best test, the SVM is trained and tested by three categories 1C, 4C and 5C based on different features as input. After optimization of the classifier parameters, the best detection rate and FPR obtained are 98.99% and 0.2 respectively. On the other hand, the last test was based on 1C and 5C categories consisted of selected the relevant features. In this stage, the SVM created the performance up to 91.49%. This result indicated that the number of the relevant features eliminated could decrease the performance exactly.

This study selected 12 relevant features from the extracted 20 features in the previous studies to improve the detection rate and to evaluate the ability of feature selection method (NBPSO). The results showed that the best performance was obtained with only the selected relevant features. Although in some combined categories they contain low number of relevant features such as C2, they achieved a reasonable performance to 91.49% detection rate and 0.3% FPR.

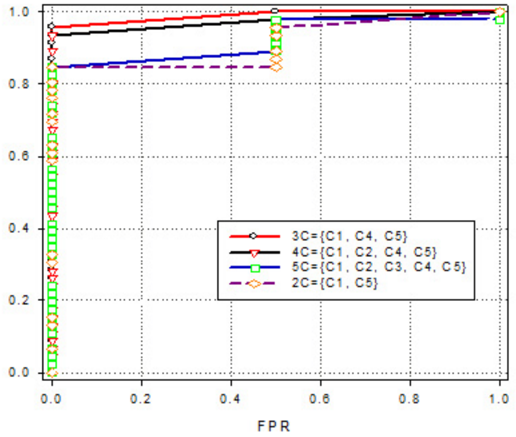


Fig. 1. ROC curve based on four (4) classifiers

5 ROC Curves and AUC Analysis

The receiver operating characteristics (ROC) graph as shown in Figure 1 illustrates classifier performance. Today, this technique has been used in machine learning because the accuracy of the classifier is not a robust measurement. ROC identifies the relationship between the false positive rate (FPR) and true positive rate (TPR). The best performance from the experiments were related to 3C, 4C, 5C, and 2C, respectively. Meanwhile, the best FPR and detection rate were 0.1% and 97.77% for 4C category. The CPU time, on the opposite, considered the impact of the number of features on the performance and computational cost. The time and cost were different by changing the number of features. Table 4 shows the changes in the CPU time and computational cost when the number of features decreased from 5 to 3, which means 63.65% to 37.2% and 215.67(s) to 173.56(s) respectively.

Table 4. The AUC results of combination features

Number of Features	AUC Value
2	94.30
3	98.21
4	98.90
5	97.68

The results in Table 4 indicated the AUC (Area Under the Curve) result is close to 1 for the 3C, 4C and 5C categories. In fact, the results showed that the best combination is related to 3C. Other combinations of features such as 2C, 4C and 5C present a good detection rate and the AUC result, although they do not contain all relevant features. Thus, the analysis presents an important point in which the categories that contain the above features may have a high detection rate and low false positive rate.

It is also noteworthy to mention that the detection rate is insufficient to assess a classifier's performance since the achieved results show that FPR has more to offer that the detection rate. This study proved this point by doing the famous statistical test, namely one-way ANOVA (Analysis of Variance). The different experiences executed based on various thresholds from 3 to 5. The F-ratio identified by ANOVA test is 9.24 ($p < 0.001$), which proved that a decrease in the detection rate is not the most important factor. Based on Table 3, eventhough 3C detects the phishing emails better than other classifiers to 98.99% of the detection rate, but the 4C classifier achieved a detection rate of 97.77% with 0.1 FPR that achieved a lower error rate in comparison with other classifiers. In addition, the best AUC of this classifier is 98.90% as compared to other classifiers.

6 Conclusions

In this study, an attempt was made to develop a spam/phishing email detection system to detect social engineering attacks. This paper proposed the New Binary Particle Swarm Optimization (NBPSO) algorithm for feature selection together with a Support Vector Machine (SVM) classifier for classification. The system was tested via a classification experiment using three datasets, namely SpamAssassin, SpamEmail, and PhishingCorpus. The experimental results, in comparison with results from previous studies, indicate that the detection system was able to reduce the number of features from 20 to 12 features, hence reducing its dimensionality. As the consequence, the accuracy rate has increased and the false positive rate (FPR) hit a lower percentage. Note that FPR represents system's reliability and has been proven by the literature that it is more important than the accuracy rate in the case of false alarm. In this work, FPR was accessed using the ROC and AUC curve.

One of the important points in classification is the parameter optimization that needs to be tuned and tested with different datasets for better classifier performance. In the future, we hope to test other datasets and to apply other metaheuristic algorithms. Comparisons will be in terms of dimensionality reduction and complexity.

Acknowledgments. This project is sponsored by the Malaysian Ministry of Higher Education (MOHE) under the Exploratory Research Grant Scheme.

References

1. Miller, T.: Social Engineering: Techniques that can Bypass Intrusion Detection Systems, `http://www.stillhq.com/pdfdb/000186/data.pdf`
2. Gorton, A.S., Champion, T.G.: Combining Evasion Techniques to Avoid Network Intrusion Detection Systems. Skaion (2004)
3. Dodge, R.C., Carver, C., Ferguson, A.J.: Phishing for User Security Awareness. Computers & Security 26(1), 73–80 (2007)
4. Hoeschele, M., Rogers, M.: Detecting Social Engineering. In: Pollitt, M., Shenoi, S. (eds.) Advances in Digital Forensics. IFIP, vol. 194, pp. 67–77. Springer, Heidelberg (2005)
5. Ashish, T.: Social Engineering: An Attack Vector Most Intricate to Tackle. Technical Report, Infosecwriters (2007)
6. Olivo, C.K., Santin, A.O., Oliveira, L.S.: Obtaining the Threat Model for E-mail Ohishing. Applied Soft Computing (2011)
7. Ruan, G., Tan, Y.: A Three-Layer Back-Propagation Neural Network for Spam Detection using Artificial Immune Concentration. Soft Computing-A Fusion of Foundations, Methodologies and Applications 14(2), 139–150 (2010)
8. Engelbrecht, A.P.: Fundamentals of Computational Swarm Intelligence, vol. 1. Wiley, London (2005)
9. El-Alfy, E.S.M., Abdel-Aal, R.E.: Using GMDH-based Networks for Improved Spam Detection and Email Feature Analysis. Applied Soft Computing 11(1), 477–488 (2011)
10. Ma, L., Ofoghi, B., Watters, P., Brown, S.: Detecting Phishing Emails using Hybrid Features. In: Symposia and Workshops on Ubiquitous, Autonomic and Trusted Computing, pp. 49–497 (2009)
11. Oliveira, A.L.I., Braga, P.L., Lima, R.M.F., Cornelio, M.L.: GA-based Method for Feature Selection and Parameters Optimization for Machine Learning Regression Applied to Software Effort Estimation. Information and Software Technology 52(11), 1155–1166 (2010)
12. Chandrasekaran, M., Narayanan, K., Upadhyaya, S.: Phishing Email Detection based on Structural Properties. In: NYS Cyber Security Conference, pp. 1–7 (2006)
13. Toolan, F., Carthy, J.: Phishing Detection using Classifier Ensembles. In: eCrime Researchers Summit, pp. 1–9 (2009)
14. Sirisanyalak, B., Sornil, O.: Artificial Immunity-based Feature Extraction for Spam Detection. In: International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, pp. 359–364 (2007)
15. Lai, C.H.: Particle Swarm Optimization-aided Feature Selection for Spam Email Classification, p. 165. IEEE, Kumamoto (2007)
16. Ramadan, R.M., Abdel-Kader, R.F.: Face Recognition using Particle Swarm Optimization-based Selected Features. International Journal of Signal Processing, Image Processing and Pattern Recognition 2(1), 51–66 (2009)
17. Guyon, I., Matic, N., Vapnik, V.: Discovering Informative Patterns and Data Cleaning. In: KDD Workshop, pp. 145–156 (1994)
18. Chen, J., Guo, C.: Online Detection and Prevention of Phishing Attacks. In: 1st Int. Conference on Communications and Networking, pp. 1–7 (2006)

19. Mukkamala, S., Sung, A.: Significant Feature Selection using Computational Intelligent Techniques for Intrusion Detection. In: Advanced Methods for Knowledge Discovery from Complex Data, pp. 285–306 (2005)
20. Macia-Perez, F., Mora-Gimeno, F., Marcos-Jorquera, D., Gil-Martinez-Abarca, J.A., Ramos-Morillo, H., Lorenzo-Fonseca, I.: Network Intrusion Detection System Embedded on a Smart Sensor. IEEE Transactions on Industrial Electronics 58(3), 722–732 (2011)

A New Hybrid Algorithm for Document Clustering Based on Cuckoo Search and K-means

Ishak Boushaki Saida[1], Nadjet Kamel[2], and Bendjeghaba Omar[3]

[1] Computer Science Department, Faculty of Sciences, University M'Hamed Bougara,
Boumerdès, Independency Avenue, 35000, LRIA (USTHB) and LIMOSE, Algeria
saida_2005_compte@yahoo.fr

[2] Univ.Setif, Fac.Sciences, Depart. Computer Sciences, Setif, Algeria, and LRIA-USTHB
nkamel@usthb.dz

[3] LREEI (UMBB), University of Boumerdes, Algeria
benomar75@yahoo.fr

Abstract. In this paper we propose a new approach for document clustering based on Cuckoo Search and K-means. Due to the random initialization of centroids, cuckoo search clustering can reach better solution but the number of iterations may increase drastically. In order to overcome this drawback, we propose to replace this randomness by k-means at the beginning step. The effectiveness of the proposed approach was tested on the benchmark extracted from Reuters 21578 Text Categorization Dataset and the UCI Machine Learning Repository. The obtained results show the efficiency of the new approach in term of reducing the number of iterations and fitness values. Furthermore, it can improve the quality of clustering measured by the famous F-measure.

Keywords: Document Clustering, Vector Space Model, Cuckoo Search, K-means, Metaheuristic, Optimization, F-measure.

1 Introduction

The task of finding the information on web databases becomes more difficult. This is due to the huge number of documents in the web. A solution to this problem is dividing data into sub groups such that the documents in the same group have the same topic. This is known can be done by what we call documents clustering. Many algorithms have been proposed to solve the problem of document clustering [1] [2]. They are classified into two major classes: the hierarchical methods and the partitioning clustering.

Since the dimension of data is large and the time complexity of hierarchical clustering is quadratic, the partitioning clustering is more suitable for clustering the documents. One of the most famous partitioning methods is k-means algorithm [3]. It is very simple and it is linear time complexity. However, the k-means suffers from the problem of random initialization which leads sometimes to poor results (local minimum). To overcome this drawback, many algorithms were proposed in order to

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_6, © Springer International Publishing Switzerland 2014

find the global optima instead of the local one. Recently, metaheuristic approaches, have received increased attention from researchers dealing with document clustering problems. In 2005, a particle swarm optimization (PSO) based document clustering system was presented by Xiaohui and all [4] to find the optimal centroids. A novel approach for clustering short-text Corpora collections based on a discrete particle swarm optimizer was proposed by Cagnina and all [5] in order to find the optimal centroids. A genetic algorithm (GA) with Simultaneous and Ranked Mutation was presented by Premalatha and Natarajan [6]. A novel document clustering based on genetic algorithms (GA) using squared distance optimization was presented by Verma and all [7]. In 2012, the problem was treated using Ants approach presented by Vaijayanthi and all [8]. More recently, different hybridizations between techniques are proposed for document clustering [9] [10].

The Cuckoo Search (CS) is a new metaheuristic algorithm [11] [12]. Recent researches show the ability of CS for solving the clustering problem and it can produce best results compared to other metaheuristic [13] [14] [15]. However, the main drawback of this method is that the number of iterations increases to find an optimal solution; this is due to the random initialization of centroids at the beginning step. For large-scale and high-dimensional data, like document clustering, the number of iterations is very important and it can seriously affect the quality of clustering. In this paper, an efficient hybrid algorithm is proposed in order to improve the performance of the CS and eliminate its drawbacks. The approach is based on the CS combined with K-means. The obtained experimental results confirm the efficiency of the proposed approach to reduce the number of iterations and to improve the quality of the clustering results.

The remaining of this paper is organized as follows: In section 2, formal definitions of document clustering is presented such document representation, the similarity metric we used and the quality measure. In section 3, we describe the steps of our proposed approach. Numerical experimentation and results are provided in Section 4. Finally, conclusion and our future work are drawn in Section 5.

2 Formal Definitions

2.1 Document Representation

The documents to be clustered must be treated through several steps in the goal to be represented with numerical values using the Vector Space Model (VSM)[16]. First of all the documents are tokenized such that all the words in the document are extracted and separated. Then, the well comment words or stop words are eliminated. After that, all words must be stemmed using the Porter Stemmer [17]. Finally, each document is represented by a vector of number. These numbers are the weight of specified terms in the collection of documents. So each document d_i is represented by a vector $\left(w_{i1} w_{i2} \ldots w_{ij} \ldots w_{im}\right)$

Where: m is the number of terms in the collection of documents.

The weight indicates the importance of the corresponding term in a document. It depends on its number of occurrences in the document or term frequency (tf), and Inverse Document Frequency (idf), that we call $(TF - IDF)$.

The weight of the term j in the document i is given by the following equation:

$$w_{ij} = tf_{ij} * idf_{ij} = tf_{ij} * \log_2\left(\frac{n}{df_{ij}}\right). \tag{1}$$

Where

tf_{ij} is the number of occurrences of the term j in the document i.

df_{ij} is the number of documents, from the collection of documents, that the term j appears in it.

n is the number of documents in the collection.

2.2 The Similarity Metric

The goal of the clustering task is to assign the documents the most similar to the same cluster. This similarity is calculated by the distance between two documents [18]. Several metrics are defined, but the most used distances for document clustering are Minkowski distances and cosine measure. Minkowski distance between the document d_i and d_j is given by the following formula:

$$D_n(d_i, d_j) = \left(\sum_{p=1}^{m} \left|d_{ip} - d_{jp}\right|^n\right)^{1/n}. \tag{2}$$

Where m is the number of features.

The two special cases of Minkowski distances which are widely used for document clustering are: the city - block, also called Manhattan distance for $n = 1$, and the Euclidian distance for $n = 2$. The cosine distance is the most popular for document clustering [19]. Given two documents d_i and d_j represented by two vectors v_i, v_j, respectively, the cosine distance is given by the following formula:

$$\cos(d_i, d_j) = \frac{v_i^t v_j}{|v_i||v_j|}. \tag{3}$$

Where $|v_i|$ is the norm of the vector v_i.

2.3 F-measure Quality Measures

To evaluate the quality of clustering results, we use the famous F-measure index [20]. It gives information about the rate of documents in the correct cluster. The F-measure value is between zero and one. Higher value of F-measure indicates good clustering quality. It is based on two notions: the precision and recall. They are given by the following equations:

$$\mathrm{Re}\,call(i,j)=\frac{n_{ij}}{n_i}\,. \tag{4}$$

$$\mathrm{Pr}\,ecision(i,j)=\frac{n_{ij}}{n_j}\,. \tag{5}$$

Where:

n_{ij} is the number of objects of class i that are in cluster j.

n_j is the number of objects in cluster j.

n_i is the number of objects in class i.

The F-measure of cluster i and cluster j is given by the following equation:

$$F(i,j)=\frac{2\,\mathrm{Re}\,call(i,j)\,\mathrm{Pr}\,ecision(i,j)}{\mathrm{Pr}\,ecision(i,j)+\mathrm{Re}\,call(i,j)}\,. \tag{6}$$

3 Clustering with Cuckoo Search + K-means

The cuckoo search is a recent metaheuristic algorithm for resolving optimization problem. It was used for clustering problem and the results were better than other metaheuristic [13] [15]. However, CS gives the global optima after a big number of iterations which needs much more times. To overcome this inconvenient, we propose in this paper a hybrid algorithm based on k-means and cuckoo search optimization.

3.1 K-means

It is one of the simplest partitional clustering algorithms. It is based on the centers of clusters. The algorithm is given by the following steps:

1. Initialize the k centers (centroids) randomly;
2. Assign each object to the nearest cluster;

3. *Calculate the centroid c_i of each cluster which will be the new centroid. It is given by the following formula:*

$$c_i = \frac{1}{n_i} \sum_{\forall d_i \in S_i} d_i \tag{7}$$

Where d_i denotes the document that belongs to the cluster S_i ; n_i is the number of documents in cluster S_i .

Repeat steps 2 and 3 until there is no change for each cluster or predefined number of iterations is achieved.

3.2 Cuckoo Search + K-means Algorithm (Our Contribution)

In the cuckoo search clustering algorithm, we initialize the initial population randomly [13], which requires a big number of iterations to converge to the optimum solution. We propose in this paper a hybrid algorithm based on k-means and cuckoo search clustering (CS+K-means). In this algorithm we initialize the population of cuckoo search clustering by the solutions produced by k-means after a small number of iterations. The detail of this algorithm is given by the following steps:

1. *Give N the size of the population;*
2. *For $i = 1$ to N*
   (i) Execute K-means algorithm predefined numbers of iterations;
   (ii) Initialize the cuckoo search module by the k centroids produced by k-means
3. *End for;*

The N initial solutions are represented by the $N * k$ centroids resulting from k-means algorithm;

4. *Calculate the fitness of these N solutions and find the best solution; such that each document in the collection is assigned to the nearest centroid and the distance between the document and the centroid is calculated using the cosine distance. The objective function, used to minimize the sum intra cluster (SSE), is given by the following formula:*

$$SSE = \sum_{i=1}^{k} \sum_{j=1}^{n} W_{ij} * \cos(c_i, d_j) \tag{8}$$

Where: $W_{ij} = 1$ *if the document* d_j *is in the cluster* c_i *and* 0 *otherwise.* k *is the number of clusters,* n *is the number of documents and* $\cos(c_i, d_j)$ *is the cosine distance between the document* d_j *and the centroid* c_i *.*

```
5. While (t ≤ MaxGeneration) or (stop criterion);
    (i) Generate $N$ new solutions with the cuckoo search using the best solution;
    (ii) Calculate the fitness of the new solutions;
    (iii) Compare the new solutions with the old solutions. If the new solution is better than the old one, replace the old solutions by the new one ;
    (iv) Generate a fraction $(pa)$ of new solutions to replace the worse solutions;
    (v) Compare these solutions with the old solutions. If the new solution is better than the old solution, replace the old solution by the new one;
    (vi) Find the best solution;
6. End while;
7. Print the best solution and fitness;
```

4 Experimentation and Results

4.1 Datasets

To test the effectiveness of our proposed approach, we have extracted two datasets from the well known corpus Reuters-21578 Text Categorization Dataset [21]. We have chosen this corpus because the topic of documents is prelabelled. The description detail of text document datasets is given in table 1. We have used a third dataset: the Wisconsin breast cancer standard dataset [22] represented in table 2.

Table 1. Summary of text document datasets

Data	Number of documents	Number of terms	Topics	Number of groups
Dataset1	100	2798	Cocoa, gas, grain, ship and sugar	5
Dataset2	240	4422	gas, grain, ship, sugar, coffee and crude	6

Table 2. Description of dataset3

Dataset3	Number of instances	Number of attributes	Number of groups
Cancer dataset	683	9	2

4.2 Related Parameters

In order to investigate the proposed approach, the probability of worse nests was set to 0.25. The population size was set to 10 for dataset1 and dataset2, and 20 for dataset3. The cosine distance is used for dataset1 and dataset2 and the Euclidian distance is used for dataset3. The number of iterations of k-means was set to 10 iterations for the proposed method.

In order to show the convergence behaviour of CS algorithm and our proposed approach, we have used the same number of iterations for the two algorithms. In the case of CS program, the population is initialised by random centers. However, in the proposed approach we have initialised the population by the centers generated by K-means.

In the other hand, to compare the performance of the proposed approach, in term of fitness and F-measure values, we have executed 100 iterations for the two methods. In the case of our proposed algorithm 10 iterations are reserved for the K-means because the K-means algorithm tends to converge faster than other clustering algorithms, and 90 iterations for CS.

4.3 Results and Comparisons

The convergence behaviour graphs obtained by the two algorithms for each dataset are given by figure 1, figure 2, and figure 3. The results of comparisons in term of fitness and F-measure are illustrated in table 3 and table 4 respectively.

As we can see from figure 1, figure 2 and figure 3, the gaps between the graphs variations obtained by CS algorithm and our proposed approach are very significant. The CS+K-means graphs are always under the CS graphs. For all the datasets it is very clear that the graphs are completely separated and no one crosses the other.

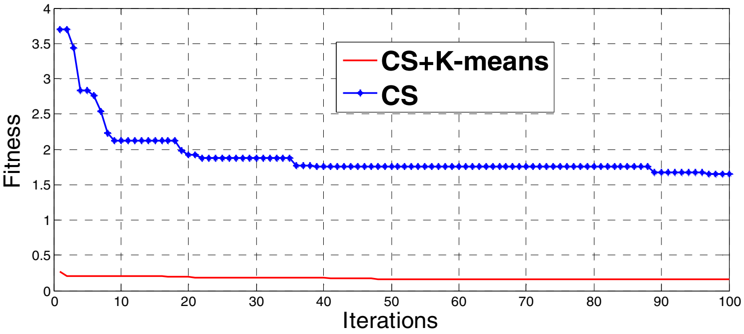


Fig. 1. The convergence behaviours of CS and CS+K-means for dataset1

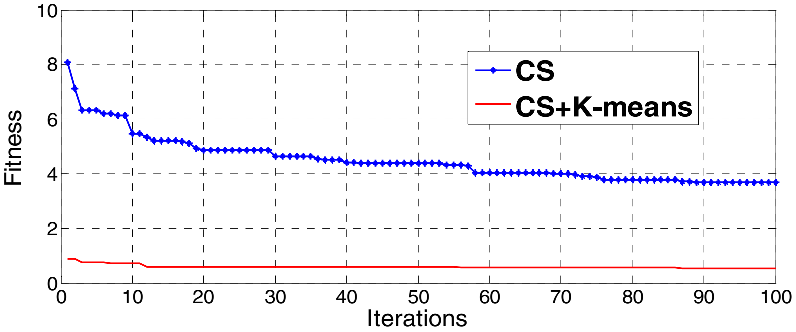


Fig. 2. The convergence behaviours of CS and CS+K-means for dataset2

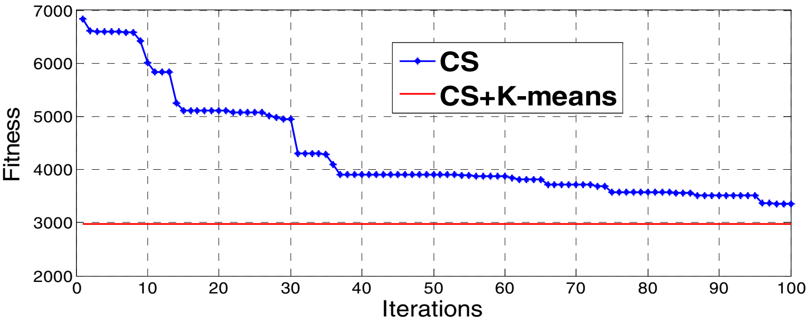


Fig. 3. The convergence behaviours of CS and CS+K-means for dataset3

As we can see from table 3, for the same iterations number, the gap between the first fitness values obtained by CS+K-means and CS is very significant for all datasets. For the dataset1, the first fitness value using the CS is 3.70, whereas this value is 0.26 for our proposed method. The best fitness value obtained by CS+K-means is much less than the corresponding one obtained by CS. For instance, the obtained best fitness value using CS for dataset1 is 1.64 and 0.15 when CS+K-means is used.

From table 4, it is obviously clear that the CS+K-means is superior then the CS in term of clustering quality measured by F-measure for the three datasets. For example, in the case of dataset1, the obtained F-measure by CS+K-means is 0.85 and 0.72 when CS is used.

Table 3. Fitness comparison of CS and CS+ K-means for the three datasets

	Number of iterations	CS		CS+K-means	
		First fitness value	Last fitness value	First fitness value	Last fitness value
Dataset1	100	3.70	1.64	0.26	0.15
Dataset2		8.06	3.67	0.89	0.52
Dataset3		6835.86	3347.73	2978.17	2965.90

Table 4. F-measure comparison of CS and CS+ K-means for the three datasets

	CS		CS+K-means	
	Number of iterations	F-measure	Number of iterations	F-measure
Dataset1	100	0.72	100	0.85
Dataset2		0.69		0.74
Dataset3		0.95		0.96

5 Conclusion

In this paper, we have proposed a new hybrid algorithm for document clustering based on cuckoo search optimization combined with k-means method. This novel approach was tested on the documents of Reuters 21578 Text Categorization Dataset and the UCI Machine Learning Repository standard dataset. The experimental results show the efficiency of the proposed approach. Compared to cuckoo search, the proposed method can reach the optima in a few iterations number with improved fitness values. Furthermore, it can improve the quality of clustering measured by F-measure. Finally, we plan as future work to analyze the sensitivity of the proposed approach to the cuckoo search control parameters for document clustering.

References

1. Patel, D., Zaveri, M.: A Review on Web Pages Clustering Techniques. In: Wyld, D.C., Wozniak, M., Chaki, N., Meghanathan, N., Nagamalai, D. (eds.) NeCoM 2011, WeST 2011, WiMoN 2011. CCIS, vol. 197, pp. 700–710. Springer, Heidelberg (2011)
2. Zamir, O., Etzioni, O.: Web Document Clustering: a Feasibility Demonstration. In: Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1998), pp. 46–54 (1998)
3. Jain, A.K.: Data Clustering: 50 Years beyond K-means. Pattern Recognition Letters 31, 651–666
4. Xiaohui, C., Thomas, E.P., Palathingal, P.: Document Clustering using Particle Swarm Optimization. In: Proceedings of the 2005 Swarm Intelligence Symposium, SIS 2005, pp. 185–191. IEEE (2005)
5. Cagnina, L., Errecalde, M., Ingaramo, D., Rosso, P.: A Discrete Particle Swarm Optimizer for Clustering Short-Text Corpora. In: Int. Conf. on Bioinspired Optimization Methods and their Applications, BIOMA 2008, pp. 93–103 (2008)
6. Premalatha, K., Natarajan, A.M.: Genetic Algorithm for Document Clustering with Simultaneous and Ranked Mutation. Modern Applied Science 3(2) (2009)
7. Verma, H., Kandpal, E., Pandey, B., Dhar, J.: A Novel Document Clustering Algorithm Using Squared Distance Optimization Through Genetic Algorithms (IJCSE) International Journal on Computer Science and Engineering 2(5), 1875–1879 (2010)
8. Vaijayanthi, P., Natarajan, A.M., Murugadoss, R.: Ants for Document Clustering. IJCSI International Journal of Computer Science Issues 9(2(2)) (March 2012)

9. Hyma, J., Jhansi, Y., Anuradha, S.: A New Hybridized Approach of PSO and GA for Document Clustering. International Journal of Engineering Science and Technology 2(5), 1221–1226 (2010)
10. Kamel, N., Ouchen, I., Baali, K.: A Sampling-PSO-K-means Algorithm for Document Clustering. In: Pan, J.-S., Krömer, P., Snášel, V. (eds.) Genetic and Evolutionary Computing. AISC, vol. 238, pp. 45–54. Springer, Heidelberg (2014)
11. Yang, X.-S., Deb, S.: Cuckoo Search via Levy Flights. In: Proc. of World Congress on Nature and Biologically Inspired Computing (NaBIC 2009), India, pp. 210–214. IEEE Publications, USA (2009)
12. Yang, X.-S., Deb, S.: Engineering Optimisation by Cuckoo Search. International Journal of Mathematical Modelling and Numerical Optimisation 1(4-30), 330–343 (2010)
13. Saida, I.B., Nadjet, K., Omar, B.: A New Algorithm for Data Clustering Based on Cuckoo Search Optimization. In: Pan, J.-S., Krömer, P., Snášel, V. (eds.) Genetic and Evolutionary Computing. AISC, vol. 238, pp. 55–64. Springer, Heidelberg (2014)
14. Zaw, M.M., Mon, E.E.: Web Document Clustering Using Cuckoo Search Clustering Algorithm based on Levy Flight. International Journal of Innovation and Applied Studies 4(1), 182–188 (2013) ISSN 2028-9324
15. Senthilnatha, J., Vipul Das, S.N., Omkar, V.: Clustering using Levy Flight Cuckoo Search. In: Bansal, J.C., et al. (eds.) Proceedings of Seventh International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2012). AISC, vol. 202, pp. 65–75. Springer, Heidelberg (2013)
16. Salton, G., Wong, A., Yang, C.S.: A Vector Space Model for Automatic Indexing. Communications of the ACM 18(11), 613–620 (1975)
17. Porter, M.F.: An Algorithm for Suffix Stripping. Program 14(3), 130–137 (1980)
18. Hammouda, K.M.: Web Mining: Clustering Web Documents A Preliminary Review. Department of Systems Design Engineering University of Waterloo, Ontario, Canada N2L 3G1 (February 26, 2001)
19. Anna, H.: Similarity Measures for Text Document Clustering. In: NZCSRSC 2008, Christchurch, New Zealand (April 2008)
20. Rendón, E., Abundez, I., Arizmendi, A., Quiroz, E.M.: Internal Versus External Cluster Validation Indexes. International Journal 5(1), 27–34 (2011)
21. David, D.L.: Test Collections Reuters-21578, `http://www.daviddlewis.com/resources/testcollections/reuters21578`
22. Merz, C.J., Blake, C.L.: UCI Repository of Machine Learning Databases, `http://www.ics.uci.edu/-mlearn/MLRepository.html`

A New Positive and Negative Linguistic Variable of Interval Triangular Type-2 Fuzzy Sets for MCDM

Nurnadiah Zamri* and Lazim Abdullah

School of Informatics and Applied Mathematics, University Malaysia Terengganu, 21030
Kuala Terengganu, Terengganu, Malaysia
nadzlina@yahoo.co.uk, lazim_m@umt.edu.my

Abstract. Fuzzy linguistic variable in decision making field has received significant attention from researchers in many areas. However, the existed research is given attention only in one side rather than two sides. Therefore, the aim of this paper is to introduce a new linguistic variable which considers both sides, positive and negative sides for symmetrical interval triangular type-2 fuzzy set (T2 FS). This new linguistic variable is developed in line with the interval type-2 fuzzy TOPSIS (IT2 FTOPSIS) method. Besides, a ranking value for aggregation process is modified to capture both positive and negative aspect for triangular. Then, this new method is tested using two illustrative examples. The results show that the new method is highly beneficial in terms of applicability and offers a new dimension to problem solving technique for the type-2 fuzzy group decision-making environment.

Keywords: Interval type-2 fuzzy sets, interval type-2 fuzzy TOPSIS.

1 Introduction

Linguistic variable is a variable whose values are not numbers but words or sentences in a natural or artificial language [1]. The concept of fuzzy linguistic variable is a staple of the type-1 fuzzy set theory [2;3]. It has the remarkable property of putting together symbols and the meaning of those symbols as proper elements of a computational system [4]. Various authors used linguistic variable methods naturally applied with type-1 fuzzy sets in many areas. For example, Cheng et al. [5] proposed a new method for evaluating weapon systems by analytical hierarchy process (AHP) based on linguistic variable weight. On the other hand, Doukas et al. [6] presented a direct and flexible multi-criteria decision making approach, using linguistic variables, to assist policy makers in formulating sustainable technological energy priorities.

Since Zadeh [1] introduced type-2 fuzzy sets, linguistic variable with type-2 fuzzy sets has spread vigorously in many areas. Such that, Wu and Mendel [7] proposed a vector similarity measure (VSM) for IT2 FSs, whose two elements measure the similarity in shape and proximity where the VSM gives more reasonable results than all other existing similarity measures for IT2 FSs for the linguistic approximation

[*] Corresponding author.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_7, © Springer International Publishing Switzerland 2014

problem. Besides, Zhoua et al. [8] defined a new type of OWA operator, the type-1 OWA operator that works as an uncertain OWA operator to aggregate type-1 fuzzy sets with type-1 fuzzy weights, which used to aggregate the linguistic opinions or preferences in human decision making with linguistic weights.

Type-2 fuzzy linguistic variables have continuously developed in many fields especially in decision making field. For examples, Chen and Lee [9] presented an interval type-2 fuzzy TOPSIS method to handle fuzzy multiple attributes group decision-making problems based on interval type-2 fuzzy sets where the weights of the attributes and ratings for alternatives were in interval type-2 fuzzy linguistic variables terms. Then, Ngan [10] extended the type-2 linguistic sets with as few postulates as possible, uniform approach for developing methodologies for fundamental tasks such as taking the union and intersection of and performing arithmetic operations on type-2 linguistic numbers. Moreover, Zhang and Zhang [11] introduced the concept of trapezoidal interval type-2 fuzzy numbers and presented some arithmetic operations between them and expressed linguistic variables assessments by transforming them into numerical variables objectively. Then, proposed trapezoidal interval type-2 fuzzy soft sets.

However, the existed research is given attention only in one side rather than two sides. Every matter has two sides such as negative and positive, bad and good and etc. [12]. This concept can be proved by Ying and Yang's theories where it is becoming one rotundity when both side turn into complementary. Besides, Zhang and Zhang [13] proved that any product can have both good and/or bad aspects. Moreover, Zadeh [2] assumed that for every non-membership degree is equal to one minus membership degree and this makes the fuzzy sets complement. The concept of two sides was successfully discussed in some of the research papers. For example, Imran et al. [12] developed a new idea condition for conflicting bifuzzy sets by relaxing the condition of intuitionistic fuzzy sets (IFS). Evaluation of both sides which were positive and negative sides in conflicting phenomena was calculated concurrently by relaxing the condition in IFS. Moreover, Nur Syibrah et al. [14] aimed to consider both sides; positive and negative by proposing the conflicting bifuzzy preference relations in group decision making by utilizing of a novel score function.

Thus, to address some of the shortcomings, this study was initiated with the idea of considering two sides which are positive and negative aspects simultaneously in the judgment process. Therefore, this paper employs a new linguistic variable which considers positive and negative aspects in term of interval type-2 fuzzy sets (IT2 FSs). The developments of this paper are threefold. Firstly, a new linguistic variable is developed which considers positive and negative aspects for interval triangular type-2 fuzzy sets (T2 FS). Secondly, a ranking value for aggregation process is modified to capture both positive and negative aspect for triangular. This study focuses only in triangular due to the suitability in defining the interval type-2 fuzzy membership function.

This paper proceeds as follows. A new linguistic variable of interval triangular type-2 fuzzy TOPSIS (IT2 FTOPSIS) is developed in Section 2. Next, full steps of IT2 FTOPSIS are shown in Section 3. In Section 4, numerical examples are presented to illustrate the proposed method. To demonstrate the feasibility and consistency of the new method, results' comparison is performed in Section 5. Finally, a simple conclusion is made in Section 6.

2 Linguistic Evaluation

A linguistic data is a variable whose value is naturally language phase in dealing with too complex situation to be described properly in conventional quantitative expressions [15]. A linguistic variable is a variable whose values are words or sentences in a natural or artificial language [1]. In today's highly competitive environment, an effective linguistic variable proofs to be an advantage and also a vital requirement in any entity. Decision makers (DMs) usually make decisions based on incomplete sources of information and this occurs due to the multiple-factors involved and these factors need to be considered simultaneously in the decision making process [16]. Imprecise sources, information ambiguity and uncertain factors are serious threats for the smooth and effective running of any entity [17]. At present, most evaluation processes are made after considering only positive aspect without considering the negative aspects. Therefore, this section introduces a new idea on linguistic variable where evaluates both positive and negative aspects.

Based on the idea by Mendel and Wu [7], we derived a new positive and negative symmetrical triangular T2FS propositions. Therefore,

Let $\tilde{\tilde{A}}$ be a positive and negative symmetrical interval triangular T2 FS as

$$\tilde{\tilde{A}} = \left(\tilde{A}_{LMF}, \tilde{A}_{UMF}\right) = \left((-b,-a),(a,b); H_1\left(\tilde{A}_{LMF}\right), H_2\left(\tilde{A}_{UMF}\right)\right), \tag{1}$$

where the value of $(-b,-a)$ is stated at the negative side of symmetrical interval triangular T2FS and (a,b) is stated at the positive side of symmetrical interval triangular T2FS.

Proposition 1
The addition operation between two positive and negative symmetrical triangular T2FS $\tilde{\tilde{A}}_1 = \left(\tilde{A}_{(1)LMF}, \tilde{A}_{(1)UMF}\right) = \left((-b_1,-a_1),(a_1,b_1); H_1\left(\tilde{A}_{(1)LMF}\right), H_2\left(\tilde{A}_{(1)UMF}\right)\right)$ and $\tilde{\tilde{A}}_2 = \left(\tilde{A}_{(2)LMF}, \tilde{A}_{(2)UMF}\right) = \left((-b_2,-a_2),(a_2,b_2); H_1\left(\tilde{A}_{(2)LMF}\right), H_2\left(\tilde{A}_{(2)UMF}\right)\right)$ is defined as follows:

$$\begin{aligned}
&\tilde{\tilde{A}}_1 \oplus \tilde{\tilde{A}}_2 \\
&= \left((-b_1,-a_1),(a_1,b_1); H_1\left(\tilde{A}_{(1)LMF}\right), H_2\left(\tilde{A}_{(1)UMF}\right)\right) \oplus \\
&\left((-b_2,-a_2),(a_2,b_2); H_1\left(\tilde{A}_{(2)LMF}\right), H_2\left(\tilde{A}_{(2)UMF}\right)\right) \\
&= \begin{pmatrix} (-b_1 \oplus -b_2, -a_1 \oplus -a_1),(a_1 \oplus a_1, b_1 \oplus b_2); \\ \left(\min\left(H\left(\tilde{A}_{(1)LMF}\right), H\left(\tilde{A}_{(2)LMF}\right)\right)\right)\left(\min\left(H\left(\tilde{A}_{(1)UMF}\right), H\left(\tilde{A}_{(2)UMF}\right)\right)\right) \end{pmatrix}
\end{aligned} \tag{2}$$

Similarly, for subtraction operation, multiplication operation, symbols of addition in addition operation and multiply in multiplication operation are changed into the subtraction and multiplication symbols.

Proposition 2
The arithmetic operation between the positive and negative symmetrical triangular T2FS $\tilde{\tilde{A}}_1 = \left(\tilde{A}_{(1)LMF}, \tilde{A}_{(1)UMF}\right) = \left((-b_1, -a_1), (a_1, b_1); H_1\left(\tilde{A}_{(1)LMF}\right), H_2\left(\tilde{A}_{(1)UMF}\right)\right)$ and the crisp value k is defined as follows:

$$k\tilde{\tilde{A}}_1 = \left(k\tilde{A}_{(1)LMF}, k\tilde{A}_{(1)UMF}\right) = \left((-b_1 \times k, -a_1 \times k), (a_1 \times k, b_1 \times k); H_1\left(\tilde{A}_{(1)LMF}\right), H_2\left(\tilde{A}_{(1)UMF}\right)\right) \tag{3}$$

$$\begin{aligned}\frac{\tilde{\tilde{A}}_1}{k} &= \left(\tilde{A}_{(1)LMF} \times \frac{1}{k}, \tilde{A}_{(1)UMF} \times \frac{1}{k}\right) \\ &= \left(\left(-b_1 \times \frac{1}{k}, -a_1 \times \frac{1}{k}\right), \left(a_1 \times \frac{1}{k}, b_1 \times \frac{1}{k}\right); H_1\left(\tilde{A}_{(1)LMF}\right), H_2\left(\tilde{A}_{(1)UMF}\right)\right)\end{aligned} \tag{4}$$

In order to make this conversion possible, positive and negative interval triangular type-2 fuzzy sets $\tilde{\tilde{A}} = \left(\tilde{A}_{LMF}, \tilde{A}_{UMF}\right) = \left((-b, -a), (a, b); H_1\left(\tilde{A}_{LMF}\right), H_2\left(\tilde{A}_{UMF}\right)\right)$ are divided into two parts $\overline{\mu}_{\tilde{A}}(x)$ and $\underline{\mu}_{\tilde{A}}(x)$, $\forall x \in X$.

where $\overline{\mu}_{\tilde{A}}(x)$ is considered as positive side and $\underline{\mu}_{\tilde{A}}(x)$ is considered as negative side.

$$\overline{\mu}_{\tilde{A}}(x) = (a, b) \text{ and } \underline{\mu}_{\tilde{A}}(x) = (-b, -a) \tag{5}$$

Here, this study uses seven basic linguistic terms as "Very Low" (VL), "Low" (L), "Medium Low" (ML), "Medium" (M), "Medium High" (MH), "High" (H),"Very High" (VH) and linguistic terms for the ratings of the criteria, which are "Very Poor" (VP), "Poor" (P), "Medium Poor" (MP), "Medium" (M)/ "Fair" (F), "Medium Good" (MG), "Good" (G) and "Very Good" (VG). Thus, the new linguistic variable of positive and negative interval triangular type-2 fuzzy sets shows as follows:

Table 1. The new linguistic variables for the relative importance weights of attributes

Linguistic Variable	Positive and Negative Interval Triangular Type-2 Fuzzy Sets
Very Low (VL)	((-0.1,-0.05),(0.05,0.1);(1)(1))
Low (L)	((-0.3,-0.1),(0.1,0.3);(1)(1))
Medium Low (ML)	((-0.5,-0.3),(0.3,0.5);(1)(1))
Medium (M)	((-0.7,-0.5),(0.5,0.7);(1)(1))
Medium High (MH)	((-0.9,-0.7),(0.7,0.9);(1)(1))
High (H)	((-1,-0.9),(0.9,1);(1)(1))
Very High (VH)	((-1,-1),(1,1);(1)(1))

Table 2. The new linguistic variables for the ratings of attributes

Linguistic Variable	Positive and Negative Interval Triangular Type-2 Fuzzy Sets
Very Poor (VP)	((-1,-0.5),(0.5,1);(1)(1))
Poor (P)	((-3,-1),(1,3);(1)(1))
Medium Poor (MP)	((-5,-3),(3,5);(1)(1))
Medium (M)/Fair (F)	((-7,-5),(5,7);(1)(1))
Medium Good (MG)	((-9,-7),(7,9);(1)(1))
Good (G)	((-10,-9),(9,10);(1)(1))
Very Good (VG)	((-10,-10),(10,10);(1)(1))

Next, we apply this new linguistic variable into the IT2 FTOPSIS. Steps are shown in Section 3.

3 An Algorithm

Suppose an IT2 FTOPSIS has n alternatives $(A_1,\ldots,A_n)$ and m decision criteria/ attributes $(C_1,\ldots,C_m)$. Each alternative is evaluated with respect to the m criterias/ attributes. All the values/ ratings assigned to the alternatives with respect to each criterion from a decision matrix, denoted by $S=(y_{ij})_{n\times m}$, and the relative weight vector about the criteria, denoted by $W=(w_1,\ldots,w_m)$, that satisfying $\sum_{j=1}^{m} w_j = 1$. Therefore, the further explanations on the proposed of the eight steps method are described as follows:

Step 1: Establish a decision matrix and weight matrix
Establish a 'positive and negative interval triangular T2 FS' decision matrix and 'positive and negative interval triangular T2 FS' fuzzy weight matrix as follows:

$$
D=\begin{array}{c} \\ s_1 \\ s_2 \\ \vdots \\ s_i \end{array}
\begin{array}{c}
\begin{array}{cccc} C_1 & C_2 & \cdots & C_j \end{array} \\
\left[\begin{array}{cccc}
\tilde{\tilde{A}}_{11}/\tilde{\tilde{B}}_{11} & \tilde{\tilde{A}}_{12}/\tilde{\tilde{B}}_{12} & \cdots & \tilde{\tilde{A}}_{1j}/\tilde{\tilde{B}}_{1j} \\
\tilde{\tilde{A}}_{21}/\tilde{\tilde{B}}_{21} & \tilde{\tilde{A}}_{22}/\tilde{\tilde{B}}_{22} & \cdots & \tilde{\tilde{A}}_{2j}/\tilde{\tilde{B}}_{2j} \\
\vdots & \vdots & \ddots & \vdots \\
\tilde{\tilde{A}}_{i1}/\tilde{\tilde{B}}_{i1} & \tilde{\tilde{A}}_{i2}/\tilde{\tilde{B}}_{i2} & \cdots & \tilde{\tilde{A}}_{ij}/\tilde{\tilde{B}}_{ij}
\end{array}\right]
\end{array},
\qquad (6)
$$

$$
\tilde{\tilde{W}}=\left(\tilde{\tilde{w}}_1/\tilde{\tilde{x}}_1,\quad \tilde{\tilde{w}}_2/\tilde{\tilde{x}}_2,\quad \cdots,\quad \tilde{\tilde{w}}_j/\tilde{\tilde{x}}_j\right)
$$

where $x_1,x_2,\ldots,x_i$ represents the alternative, $C_1,C_2,\ldots,C_j$ represents the attribute and $\tilde{\tilde{W}}$ represents the weight. Each entry value considered as 'positive and negative interval triangular T2 FS' values, which denoted as $\tilde{\tilde{A}}_{ij}$ ('positive and negative interval triangular T2 FS') and $\tilde{\tilde{w}}_j$ ('weight of positive and negative interval triangular T2 FS').

Step 2: Weighting matrix

Construct the weighting matrix W_p of the attributes of the decision-maker and construct the *p*th average weighting matrix $\overline{W}$, respectively, shown as follows:

$$\overline{W}_p = \left(\tilde{\tilde{w}}_j^{\,p} / \tilde{\tilde{x}}_j^{\,p}\right)_{1\times m} = \begin{array}{c} \begin{array}{cccc} f_1 & f_2 & \cdots & f_n \end{array} \\ \left[\begin{array}{cccc} \tilde{\tilde{w}}_1^{\,p} / \tilde{\tilde{x}}_1^{\,p} & \tilde{\tilde{w}}_2^{\,p} / \tilde{\tilde{x}}_2^{\,p} & \cdots & \tilde{\tilde{w}}_j^{\,p} / \tilde{\tilde{x}}_j^{\,p} \end{array}\right] \end{array} \tag{7}$$

$$\overline{W} = \left(\tilde{\tilde{w}} / \tilde{\tilde{x}}\right)_{1\times m} \tag{8}$$

where $\tilde{\tilde{w}}_j / \tilde{\tilde{x}}_j = \dfrac{\tilde{\tilde{w}}_j^1 / \tilde{\tilde{x}}_j^1 \oplus \tilde{\tilde{w}}_j^2 / \tilde{\tilde{x}}_j^2 \oplus \ldots \oplus \tilde{\tilde{w}}_j^k / \tilde{\tilde{x}}_j^k}{k}$, $\tilde{\tilde{w}}_j / \tilde{\tilde{x}}_j$ is a 'positive and negative interval triangular, $1 \le j \le m$, $1 \le p \le k$ and denotes the number of decision-makers.

Step 3: Weighted decision matrix

Construct the weighted decision matrix $\overline{Y}_w$.

Step 4: Calculate the ranking value

To aggregate the judgment matrices into IT2 FS, the concept of ranking trapezoidal IT2 FS proposed by Xu [18] is used. Modification to the matrices is also made by taking directly values of weighted decision matrix for positive and negative interval T2 FS. The further calculation is shown in the next section.

Let $\tilde{\tilde{A}} = \left(\tilde{A}_{LMF}, \tilde{A}_{UMF}\right) = \left((-b,-a),(a,b); H_1\left(\tilde{A}_{LMF}\right), H_2\left(\tilde{A}_{UMF}\right)\right)$ be positive and negative interval triangular T2 FS. According to the weighted decision matrix, corresponding ranking values of alternatives can be determined based on the following equations:

The ranking value of the negative interval triangular T2 FS $\tilde{A}^+$ of the positive and negative interval triangular T2 FS $\tilde{A}$ also can be calculated as follows:

$$Rank\left(\tilde{\tilde{A}}^+\right) = \left|\frac{1}{n(n-1)}\left(\sum_{j=1}^{n} p_{ij}\left(\tilde{\tilde{A}}^+\right) + \frac{n}{2} - 1\right)\right| \tag{9}$$

$$\text{where } 1 \le i \le n \text{ and } \left|\sum_{i=1}^{n} Rank\left(\tilde{\tilde{A}}^+\right)\right| = |1| \tag{10}$$

In the same way, the ranking value of the negative interval triangular T2 FS $\tilde{A}^-$ of the positive and negative interval triangular T2 FS $\tilde{A}$ also can be calculated as follows:

$$Rank\left(\tilde{\tilde{A}}^-\right)=\left|\frac{1}{n(n-1)}\left(\sum_{j=1}^{n}p_{ij}\left(\tilde{\tilde{A}}^-\right)+\frac{n}{2}-1\right)\right| \tag{11}$$

where $1 \le i \le n$ and $\left|\sum_{i=1}^{n} Rank\left(\tilde{\tilde{A}}^-\right)\right| = |1|$ (12)

Then, the ranking value of the positive and negative interval type-2 fuzzy sets $\tilde{\tilde{A}}$ can be calculated as follows:

$$Rank\left(\tilde{\tilde{A}}\right)=\left|Rank\left(\tilde{A}^+\right)+Rank\left(\tilde{A}^-\right)\right| \tag{13}$$

where $1 \le i \le n$ and $\left|\sum_{i=1}^{n} Rank\left(\tilde{\tilde{A}}\right)\right| = |1|$. (14)

Step 5: Normalized the ranking value

The ranking values are normalized via the following equation:

$$C\left(\tilde{\tilde{v}}\right)=\frac{Rank\left(\tilde{\tilde{v}}\right)_{ij}}{\sum_{j=1}^{n} Rank\left(\tilde{\tilde{v}}\right)_{ij}}$$

where $C\left(\tilde{\tilde{v}}\right)$ is a normalized ranking value.

Step 6: Rank the values

Sort the values of $C\left(\tilde{\tilde{v}}\right)$ in a descending sequence. The larger the value of $C\left(\tilde{\tilde{v}}\right)$, the higher the preference of the alternatives s_j.

In this IT2 FTOPSIS framework, we introduced a new and standardized linguistic variable based on ‘positive and negative interval T2 FS’. This new linguistic variable is hoped to be one of standard scales while using in IT2 FTOPSIS process. Besides, the steps to calculate ranking values have been upgraded in term of ‘positive and negative interval T2 FS’ approach too.

4 Illustrative Examples

We offer two different of examples in this section which are the examples from Chen [19] and Chen and Lee [20].

Example 1 [19]
Assume that there are three decision-makers D_1, D_2 and D_3 of a software company to hire a system analysis engineer and assume that there are three alternatives x_1, x_2, x_3 and five attributes ‘‘Emotional Steadiness”, ‘‘Oral Communication Skill”, ‘‘Personality”, ‘‘Past Experience”, ‘‘Self-Confidence”.

Example 2 [20]
The example involved determining a suitable car selection among three different cars stated as $\{x_1, x_2, x_3\}$. This example has pointed three experts $\{D_1, D_2, D_3\}$ to evaluate cars and consider four attributes “Safety”, “Price”, “Appearance”, and “Performance. Let F be the set of attributes, where F={Safety (f_1), Price (f_2), Appearance (f_3), Performance (f_4)}.

After a hard effort of consideration, three experts from both examples described the information of selecting alternatives with respect to attributes using the linguistic variable in Table 1. Then, they described the evaluating values of the weight attributes using Table 2. Result is concluded in Section 5.

5 Results Validation

The comparison between two numerical examples which are Chen [19] and Chen and Lee [20] are summarized in this section. From the earlier discussions, this paper has introduced new linguistic variable where this new linguistic variable is positive and negative interval triangular T2 FS linguistic variable. Therefore, the summary of the new ranking order of the alternatives of MCDM problems using the proposed IT2 FTOPSIS is given in Table 3.

Table 3. Ranking of the problems under different methods

Example	Evaluation	Ranking Values	Rank of the Proposed Method
Example 1	Chen [19]	$x_1 = 0.19$ $x_2 = 1$ $x_3 = 0.56$	$x_2 > x_3 > x_1$
	Our proposed method	$x_1 = 0.1258$ $x_2 = 0.1601$ $x_3 = 0.1495$	$x_2 > x_3 > x_1$
Example 2	Chen and Lee [20]	$x_1 = 0.61$ $x_2 = 0.87$ $x_3 = 0.31$	$x_2 > x_1 > x_3$
	Our proposed method	$x_1 = 0.1193$ $x_2 = 0.1197$ $x_3 = 0.1168$	$x_2 > x_1 > x_3$

In Chen [19]'s example, three decision-makers D_1, D_2 and D_3 of a software company were chosen to hire a system analysis engineer. Then, three different alternatives which were x_1, x_2, x_3 and five attributes which were 'Emotional Steadiness', 'Oral Communication Skill', 'Personality', 'Past Experience' and 'Self-Confidence' were considered in order to choose a suitable system analysis engineer. It is better to note that the result from the proposed method is consistent with the existed examples.

Whereas, in Chen and Lee [20]'s example, three decision makers D_1, D_2 and D_3 were hired to evaluate the best car among three cars x_1, x_2, x_3. Four attributes were stated as 'Safety', 'Price', 'Appearance' and Performance were used in order to select the best alternatives. We can see that, all the ranking orders from the proposed method are consistent with the existed examples.

6 Conclusion

Linguistic variable delineate a useful tool in expressing decision makings (DM's) evaluations over alternatives. In this paper we have critically introduced a new linguistic variable that considers both sides which are positive and negative sides. From both sides, a new linguistic variable of 'positive and negative symmetrical interval triangular' in term of T2 FS were successfully proposed. Then, both linguistic variables were applied into IT2 FTOPSIS. Moreover, a ranking value for aggregation process is modified in line with the new linguistic variable. In order to check the efficiency of the new method, two illustrative examples from Chen [19] and Chen and Lee [20] were tested. The efficiency of using new method is proven with a straight forward computation in illustrative examples. This approach is seen to provide a new perspective in type-2 decision making area. They offer a practical, effective and simple way to produce a comprehensive judgment, which can considers various factors in the selection process, including tangibles and intangibles factors, of the real-life decision makings that are always associated with some degrees of uncertainty. Lastly, this research can further be extended by using non-symmetrical interval triangular T2 FS.

References

1. Zadeh, L.A.: The Concept of a Linguistic Variable and its Application to Approximate Reasoning. Part I. Information Sciences. 8, 199–249 (1975a)
2. Zadeh, L.A.: Fuzzy Sets. Information and Control 8, 338–353 (1965)
3. Zadeh, L.A.: Is there a Need for Fuzzy Logic? Information Sciences 13, 2751–2779 (2008)
4. Zadeh, L.A.: Toward a Generalized Theory of Uncertainty (GTU) – An outline. Information Sciences 172(1-2), 1–40 (2005)
5. Cheng, C.-H., Yang, K.-L., Hwang, C.-L.: Theory and Methodology Evaluating Attack Helicopters by AHP based on Linguistic Variable Weight. European Journal of Operational Research 116, 423–435 (1999)

6. Doukas, H.C., Andreas, B.M., Psarras, J.E.: Multi-Criteria Decision Aid for the Formulation of Sustainable Technological Energy Priorities using Linguistic Variables. European Journal of Operational Research 182, 844–855 (2007)
7. Wu, D., Mendel, J.M.: A Vector Similarity Measure for Linguistic Approximation: Interval Type-2 and Type-1 Fuzzy Sets. Information Sciences 178, 381–402 (2008)
8. Zhou, S.-M., Chiclana, F., John, R.I., Garibaldi, J.M.: Type-1 OWA Operators for Aggregating Uncertain Information with Uncertain Weights Induced by Type-2 Linguistic Quantifiers. Fuzzy Sets and Systems 159, 3281–3296 (2008)
9. Chen, S.-M., Lee, L.-W.: Fuzzy Multiple Attributes Group Decision-Making based on the Ranking Values and the Arithmetic Operations of Interval Type-2 Fuzzy Sets. Expert Systems with Applications 37, 824–833 (2010a)
10. Ngan, S.-C.: A Type-2 Linguistic Set Theory and its Application to Multi-Criteria Decision Making. Computers & Industrial Engineering 64, 721–730 (2013)
11. Zhang, Z., Zhang, S.: A Novel Approach to Multi Attribute Group Decision Making based on Trapezoidal Interval Type-2 Fuzzy Soft Sets. Applied Mathematical Modelling 37, 4948–4971 (2013)
12. Imran, C.T., Syibrah, M.N., Mohd Lazim, A.: New Condition for Conflicting Bifuzzy Sets based on Intuitionistic Evaluation. World Academy of Science, Engineering and Technology 19, 451–455 (2008)
13. Zhang, W.R., Zhang, L.: Yin-Yang Bipolar Logic and Bipolar Fuzzy Logic. Information Sciences 165, 265–287 (2004)
14. Nur Syibrah, M.N., Mohd Lazim, A., Che Mohd Imran, C.T., Abu Osman, M.T.: New Fuzzy Preference Relations and its Application in Group Decision Making. World Academy of Science, Engineering and Technology 54, 690–695 (2009)
15. Zhang, S.F., Liu, S.Y.: A GRA-Based Intuitionistic Fuzzy Multi-Criteria Group Decision Making Method for Personnel Selection. Experts Systems with Application 38, 11401–11405 (2011)
16. Zamali, T., Abu Osman, M.T., Mohd Lazim, A.: Equilibrium Linguistic Computation Method for Fuzzy Group Decision-Making. Malaysian Journal of Mathematical Sciences 6(2), 225–242 (2012)
17. Zamali, T., Lazim, M.A., Abu Osman, M.T.: Sustainable Decision-Making Model for Municipal Solid-Waste Management: Bifuzzy Approach. Journal of Sustainability Science and Management 7(1), 56–68 (2012)
18. Xu, Z.S.: A Ranking Arithmetic for Fuzzy Mutual Complementary Judgment Matrices. Journal of Systems Engineering 16(4), 311–314 (2001)
19. Chen, C.T.: Extension of the TOPSIS for Group Decision Making under Fuzzy Environment. Fuzzy Sets and Systems 114(1), 1–9 (2000)
20. Chen, S.-M., Lee, L.-W.: Fuzzy Multiple Attributes Group Decision-Making based on the Interval Type-2 TOPSIS Method. Journal of Expert Systems with Application 37, 2790–2798 (2010)

A New Qualitative Evaluation for an Integrated Interval Type-2 Fuzzy TOPSIS and MCGP

Nurnadiah Zamri* and Lazim Abdullah

School of Informatics and Applied Mathematics, University Malaysia Terengganu,
21030 Kuala Terengganu, Terengganu, Malaysia
nadzlina@yahoo.co.uk, lazim_m@umt.edu.my

Abstract. Sometimes, information needed an objectively evaluation. It is hard to determine the value of some parameters because of their uncertain or ambiguous nature. However, most of the study neglected the qualitative evaluation. This paper aims to propose a new qualitative evaluation which considers three different aspects which are linguistic to crisp, the unconvinced decision and in between. This new qualitative evaluation is developed to produce an optimal preference ranking of an integrated fuzzy TOPSIS and multi-choice goal programming MCGP in interval type-2 fuzzy sets (IT2 FSs) aspects. An example is used to illustrate the proposed method. The results show that the qualitative evaluation in the new method is suitable for the integrated interval type-2 fuzzy TOPSIS and MCGP. Results are consistent with the numerical example. This new method offers a new dimension to type-2 fuzzy group decision-making environment.

Keywords: Qualitative evaluation, interval type-2 fuzzy TOPSIS, multi-choice goal programming.

1 Introduction

Due to its wide applicability and ease of use, the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) can be divided into three commonly classes; TOPSIS, fuzzy TOPSIS and interval type-2 fuzzy TOPSIS. Recently, it is observed that the focus has been confined to the applications of the (integrated TOPSISs; integrated fuzzy TOPSISs; integrated interval type-2 fuzzy TOPSIS) rather than the stand-alone ones (TOPSIS; fuzzy TOPSIS; interval type-2 fuzzy TOPSIS).

Moreover, the integrated TOPSISs, the integrated fuzzy TOPSISs and the integrated interval type-2 fuzzy TOPSIS can also be applied in various tools. Some of the tools are integrated with analytic hierarchy process (AHP) ([1]), followed by fuzzy analytic hierarchy process (FAHP) [2], Quality Function Deployment (QFD) [3], DEA [4], TAGUCHI'S LOSS FUNCTION-multi-criteria goal programming (MCGP) [5], inexact mixed integer linear programming (IMILP) [6], Consistent Fuzzy Preference Relations (CFPR) [7], Interactive resolution method [8],

* Corresponding author.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_8, © Springer International Publishing Switzerland 2014

MOLP [9], Shannon's entropy [10], Neural network [11], Efficient version of epsilon constraint method [12], Epsilon constraint method [12], and etc.

It is observed that there are rooms for improvement in some approaches. For example, for the integrated of FTOPSIS and multi-criteria goal programming (MCGP) approach, the evaluation criteria in supplier case study used in MCGP are all in quantitative, such as budget, delivery-time, demand and supplier capacity. Some qualitative factors, such as reliability and flexibility were neglected. Besides, quality, capacity, delivery punctuality to objectively select the suppliers is unavailable or it at best imprecise [13]. Therefore, this study was initiated by developing qualitative evaluation simultaneously in the integrated process. The objective of this paper is to propose a new qualitative evaluation which considers three different aspects; linguistic to crisp, the unconvinced decision and in between. We developed a new qualitative evaluation to produce an optimal preference ranking of the integrated fuzzy TOPSIS and MCGP in interval type-2 fuzzy sets (IT2 FSs) aspects. Therefore, the developments of this paper are twofold. Firstly, a new qualitative evaluation is developed. Secondly, a new integrated fuzzy TOPSIS and MCGP in IT2 FSs terms is performed.

In the recent years, there are lots of papers discussed on an interval type-2 fuzzy TOPSIS (IT2 FTOPSIS) but too little attention has been paid to integrate IT2 FTOPSIS and MCGP. Some of them are Chen and Lee [14] presented an IT2 FTOPSIS method to handle fuzzy multiple attributes group decision-making problems based on interval type-2 fuzzy sets where the weights of the attributes and ratings for alternatives were in interval type-2 fuzzy linguistic variables terms. Besides, Chen and Lee [15] presented a new method to handle fuzzy multiple attributes group decision-making problems based on the ranking values and the arithmetic operations of interval type-2 fuzzy sets. Next, Celik et al. [16] proposed an integrated of IT2 FTOPSIS with grey relational analysis (GRA) to improve customer satisfaction in public transportation for Istanbul. To date, none paper on integrated IT2 FTOPSIS with MCGP model.

For these reasons, integrated IT2 FTOPSIS and MCGP is developed to considers qualitative evaluation. It has been found to ease the difficultness in determining the value of some parameters/ values because of their uncertain or ambiguous nature. This approach is seen to provide a new perspective in fuzzy type-2 decision making environment. They offer a practical, effective and low risk computation to produce a comprehensive judgment.

2 Qualitative Evaluation

This section introduces the basic definitions relating to qualitative evaluation [17]. The triangular fuzzy numbers (TFNs) are attributed to linguistic terms. TFNs cannot be used directly in the decision matrix of TOPSIS. So they have to be converted to crisp numbers by using the crisp value of TFN *A*. Thus, (a, b, c) is obtained for the DMs confidence level (α) as follows:

$$I^{\alpha} = \alpha \cdot I_R\left(\tilde{A}\right) + (1-\alpha) \cdot I_L\left(\tilde{A}\right)$$

$$\begin{aligned} &= \alpha\int_0^1 f_A^R(y)dy + (1-\alpha)\int_0^1 f_A^L(y)dy \\ &= \alpha \int_0^1 \left[c-(c-b)y\right]dy + (1-\alpha)\int_0^1 \left[a+(b-a)y\right], dv \end{aligned} \tag{1}$$

Where $I_R(\tilde{A})$ and $I_L(\tilde{A})$ are the integrals of right and left areas of membership function A; f_A^R and f_A^L are the invers functions of triangular membership functions. A triangular membership function is defined as:

$$\mu(x) = \begin{cases} 0 & x \le a \\ \dfrac{x-a}{b-a} & a \le x \le b \\ \dfrac{c-x}{c-b} & b \le x \le c \\ 1 & x \ge 0 \end{cases} \tag{2}$$

The new qualitative evaluation is successfully explained in this section.

3 The Proposed Method

The main objective in this proposed method is to develop the qualitative evaluation. This qualitative evaluation is developed for MCGP method, while this MCGP method is integrated with fuzzy TOPSIS. The integrated fuzzy TOPSIS with MCGP is established in the interval type-2 fuzzy sets (IT2FSs) concept. This proposed method is seen not only considers decision-makers' preference and experience but also includes various tangible constraints, for example, the buyer's budget, supplier's capacity and delivery time.

3.1 The Proposed of Qualitative Evaluation

In the real world, it is hard to determine the value of some parameters/ values because of their uncertain or ambiguous nature. Therefore, linguistic variable defines by decision makers (DMs) is used in expressing their evaluations with imprecise terms such as "Very Poor" or "Medium". There are some factors that can be named as qualitative criteria of design alternatives such as human factors, ergonomics, serviceability, aesthetics and ease of use.

First, we divided the qualitative evaluation into three parts. First, a single terms of trapezoidal interval type-2 fuzzy number such as "Good" or "Very Good". Second, the unconvinced decision such as either "Good" or "Very Good" or can be stated as G/VG. Next, for the third part, is in between such as "from Good to Very Good" or can be stated as G ≤ y ≤ VG.

The reason why we divided the qualitative evaluation into three parts is because sometime decision-makers cannot give an exact solution for certain decision. They maybe give, 'the unconvinced decision' or 'in between' due to their experts.

Therefore, this study uses the trapezoidal interval type-2 fuzzy numbers as the linguistic terms. This trapezoidal interval type-2 fuzzy numbers cannot be used directly in the MCGP method, so they have to be converted to crisp number. Therefore, we modified some of the Liao and Wang's method (as Equation 1) to overcome this problem.

A) Trapezoidal interval type-2 fuzzy number change into crisp number

Liao and Wang [17] converted the fuzzy number into crisp number by using the triangular fuzzy number. However, in this study, we used the trapezoidal case. Therefore, first, we needed to convert the trapezoidal to triangular case.

Let, *A=(a, b, c)* be a triangular fuzzy number and *B=(d, e, f, g)* be a trapezoidal fuzzy number.

First, $\frac{e+f}{2}=h$ (3)

Then, let $h=b$

Therefore $a=d,\quad b=h,\quad c=g$ (4)

Thus, we have,

$$\therefore\ =\frac{1}{2}\left[\alpha\cdot c+b+(1-\alpha)\cdot a\right]=\frac{1}{2}\left[\alpha\cdot .g+h+(1-\alpha)\cdot d\right] \quad (5)$$

For example, we applied the trapezoidal interval type-2 fuzzy number using this new converter. Thus, we use the linguistic variable as stated in Table 1. We considers VG linguistic as an example. Thus, VG = ((0.9, 1.0, 1.0, 1.0; 1, 1); (0.9, 1.0, 1.0, 1.0; 1, 1)).

Table 1. Linguistic terms of the attributes

Linguistic Terms	Type-1 Fuzzy Sets
Very Poor (VP)	((0, 0, 0, 1; 1, 1); (0, 0, 0, 1; 1, 1))
Poor (P)	((0, 0.1, 0.1, 0.3; 1, 1); (0, 0.1, 0.1, 0.3; 1, 1))
Medium Poor (MP)	((0.1, 0.3, 0.3, 0.5; 1, 1); (0.1, 0.3, 0.3, 0.5; 1, 1))
Fair (F)	((0.3, 0.5, 0.5, 0.7; 1, 1); (0.3, 0.5, 0.5, 0.7; 1, 1))
Medium Good (MG)	((0.5, 0.7, 0.7, 0.9; 1, 1); (0.5, 0.7, 0.7, 0.9; 1, 1))
Good (G)	((0.7, 0.9, 0.9, 1.0; 1, 1); (0.7, 0.9, 0.9, 1.0; 1, 1))
Very Good (VG)	((0.9, 1.0, 1.0, 1.0; 1, 1); (0.9, 1.0, 1.0, 1.0; 1, 1))

First, assign that a_{i1}^{L} = 0.9, a_{i2}^{L} = 1.0, a_{i2}^{L} = 1.0, a_{i4}^{L} =1.0 and a_{i1}^{U} = 0.9, a_{i2}^{U} = 1.0, a_{i3}^{U} = 1.0, a_{i4}^{U} = 1.0

Let $\frac{a_{i2}^{L}+a_{i3}^{L}}{2}=\frac{1.0+1.0}{2}=1.0$ and $\frac{a_{i2}^{U}+a_{i3}^{U}}{2}=\frac{1.0+1.0}{2}=1.0$

$$=\frac{\left(\frac{1}{2}\left[\alpha \cdot a_{i4}^{L}+a_{i5}^{L}+(1-\alpha)\cdot a_{i1}^{L}\right]\right)+\left(\frac{1}{2}\left[\alpha \cdot .a_{i4}^{U}+a_{i5}^{U}+(1-\alpha)\cdot a_{i1}^{U}\right]\right)}{2} \tag{6}$$

Assume that $\alpha = 0.95$, then $\frac{0.9975+0.9975}{2} = 0.9975$

B) The unconvinced decision

For the unconvinced decision such as either "Good" or "Very Good" or can be stated as G/VG. The calculations are shown as follows:

First we add these two G and VG;

G = ((0.7, 0.9, 0.9, 1.0; 1, 1); (0.7, 0.9, 0.9, 1.0; 1, 1))
VG = ((0.9, 1.0, 1.0, 1.0; 1, 1); (0.9, 1.0, 1.0, 1.0; 1, 1))
Total = ((1.6, 1.9, 1.9, 2.0; 1, 1); (1.6, 1.9, 1.9, 2.0; 1, 1))

Next, divide into two;

$$\frac{((1.6, 1.9, 1.9, 2.0; 1, 1); (1.6, 1.9, 1.9, 2.0; 1, 1))}{2} = ((0.8, 0.95, 0.95, 1.0; 1, 1); (0.8, 0.95, 0.95, 1.0; 1, 1))$$

Assume that $\alpha = 0.95$, then $\frac{0.9925+0.9975}{2} = 0.995$ (7)

C) In between

For the in between such as "from Very Low to Medium" or can be stated as VL ≤ y ≤ M.

VL = ((0, 0, 0, 1; 1, 1); (0, 0, 0, 1; 1, 1))

Assume that $\alpha = 0.95$, then $\frac{0.475+0.475}{2} = 0.475$

Next, for M = ((0.3, 0.5, 0.5, 0.7; 1, 1); (0.3, 0.5, 0.5, 0.7; 1, 1))
Assume that $\alpha = 0.95$, then $\frac{0.59+0.59}{2} = 0.59$

Then, VL ≤ y ≤ M = $0.475 \le y \le 0.59$. (8)

The new qualitative evaluation of trapezoidal interval type-2 fuzzy number is an analogous concept of integrating IT2 FTOPSIS in MCGP procedures.

3.2 A Method of Integrated IT2 FTOPSIS in MCGP

On the basis of the earlier theoretical analysis, an approach of integrated IT2 FTOPSIS in MCGP is developed. The new concept of qualitative evaluation in the form of IT2 FSs is set and later be translated into integrated IT2 FTOPSIS in MCGP.

Assume that there is a set *X* of alternatives, where $X = \{x_1, x_2, \ldots, x_n\}$, and assume that there is a set F attributes, where $F = \{f_1, f_2, \ldots, f_m\}$. Assume that there are k decision-makers $D_1, D_2, \ldots,$ and D_k. The set F of attributes can be divided into two sets F_1 and F_2, F_1 where denotes the set of benefit attributes, F_2 denotes the set of cost attributes, F_2 and . The proposed method is now presented as follows:

Step 1: Establish a decision matrix
Construct the design matrix Y_p of the *p*th decision-maker and construct the average decision matrix respectively.

Step 2: Calculate the weighting process
Construct the weighting matrix W_p of the attributes of the decision-maker and construct the *p*th average weigthing matrix $\overline{W}$, respectively.

Step 3: Construct the weighted DMs' matrix.
Construct the weighted decision matrix $\overline{Y}_w$,

Step 4: Calculate the ranking value, $Rank\left(\tilde{\tilde{d}}_{ij}\right)$**.**

Construct the matrix for each element of IT2 FS in weighted DMs' matrix.

Step 5: Normalized the ranking value.
The ranking values of IT2 FS are normalized via the following equation to obtain weight relative:

Step 6: Calculate the relative weight of priority and rank all the alternatives.
Computing the relative weight and ranking the alternatives.

Step 7: Build the MCGP model
According to the closeness coefficients obtained from Step 6, then, build the MCGP model.

In this new framework, we introduced a new integrated IT2 FTOPSIS in MCGP. Both IT2 FTOPSIS and MCGP method considered qualitative evaluation in every step as shown in Fig. 1. This new integrated IT2 FTOPSIS in MCGP is hoped to be one of the standard method while in integrated interval type-2 fuzzy TOPSIS process.

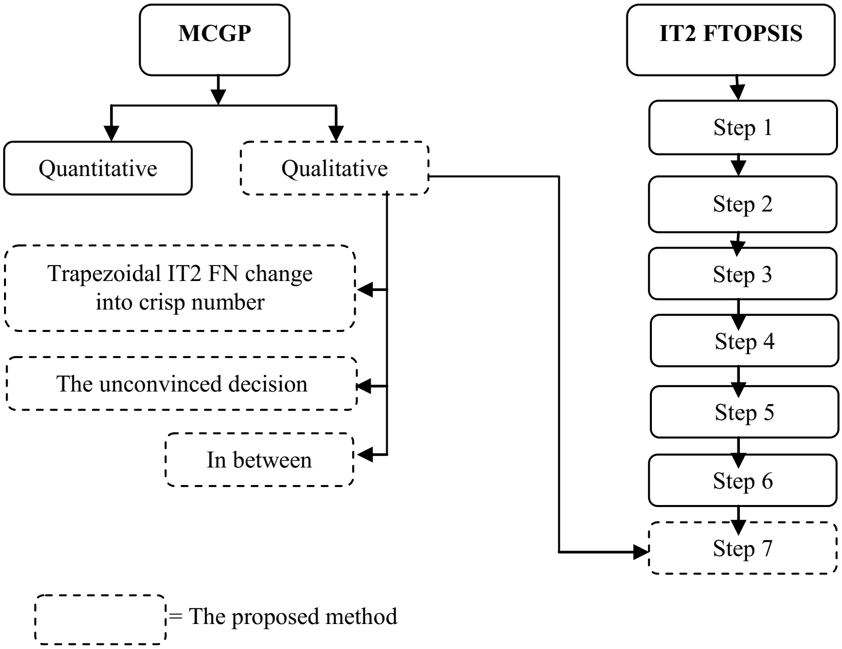


Fig. 1. Conceptual framework

4 Numerical Example

To illustrate the procedures and feasibility of the proposed integrated IT2 FTOPSIS in MCGP framework a published example are cited and used without modification. The calculation had been made using recently example from Liao and Kao [18]. The MCDM problem related to select a best supplier from four qualified suppliers (S_1, S_2, S_3, S_4). A decision committee including three decision-makers (D_1, D_2, D_3) has been formed to select the best supplier based on a complete set of criteria; Relationship closeness (C_1), Quality of product (C_2), Delivery capabilities (C_3), Warranty level (C_4) and Experience time (C_5).

As the result, the order of rating among those alternatives is $A_1 > A_3 > A_2 > A_4$. A_1 (0.25001993) is ranked first, followed by A_3 (0.249997292), A_2 (0.249995872). A_4 (0.249986905) is ranked last. After taking into account the five criteria and the opinion from three experts, a single measurement for each causes are obtained and A_1 recorded the highest closeness coefficient at 0.25001993. These results are consistent with the Liao and Kao [18]'s results.

According to the results of the closeness coefficients for each supplier, build the MCGP model to identify the best suppliers and optimum order qualities. Supplier weights (or priority values) are used as closeness coefficients in an objective function to allocate order quantities among suppliers such that the total value of procurement (TVP) is maximized.

According to the sales record in the last 5 years and the sales forecast by Formosa Watch Co., Ltd. (FCWL), the CEO and top managers of FWCL have established four goals as follows:

1) The TVP of at least 3500 units from procurement; and the more the better.
2) The total cost of procurement of less than 53200 thousand dollars; and the less the better.
3) For achieving the procurement levels, the delivery time (per batch) from supplier is set between 4 and 7 days; the less the better.
4) For seeking differentiation strategy (i.e., quality leadership), maintain the current procurement level of less than 5000 units.

In addition, the coefficients of variables in model are given by FCWL's database calculated from the last 5 years record. The unit material cost for suppliers S$_1$, S$_2$, S$_3$ and S$_4$ are \$12, \$9, \$15 and \$6, respectively, and the capacities of the four candidate suppliers S$_1$, S$_2$, S$_3$ and S$_4$ are 2700, 3500, 2300 and 3100 units, respectively. Furthermore, the delivery time levels of the four candidate suppliers are 2.5, 4, 6, and 3 days, respectively.

Three different types of qualitative evaluation are constructed using the MCGP model. Therefore, the functions and parameters related to FCWL's supplier selection problem for each type are listed follows:

We use the VG linguistic variable as stated in Equation 6, G/VG linguistic variable as stated in Equation 7 and VL $\leq$ y $\leq$ M linguistic variable as stated in Equation 8. Thus, the functions and parameters in MCGP model is calculated as follows:

$Min\, z = d_1^+ + d_1^- + d_2^+ + d_2^- + d_3^+ + d_3^- + d_4^+ + d_4^- + e_1^+ + e_1^- + e_2^+ + e_2^- + e_3^+ + e_3^-$

s.t. $0.558x_1 + 0.502x_2 + 0.516x_3 + 0.476x_4 - d_1^+ + d_2^- = y_1$

(procurement cost goal; the less the better)

$y_1 - e_1^+ + e_1^- = 46000$ for $\left|y_1 - g_{1,\min}\right|$

$46000 \leq y_1 \leq 53200$ for bound y_1

$2.5x_1 + 4x_2 + 6x_3 + 3x_4 - d_3^+ + d_3^- = y_2$ for delivery time goal

$y_2 - e_2^+ + e_2^- = 4$ for $\left|y_2 - g_{2,\min}\right|$

$4 \leq y_2 \leq 7$ for bound y_2

$x_1 + x_2 + x_3 + x_4 - d_4^+ + d_4^+ \leq 5000$ for procurement level

$x_1 \leq 2700$ for the capacity bound of S$_1$

$x_2 \leq 3500$ for the capacity bound of S$_2$

$x_3 \leq 2300$ for the capacity bound of S$_3$

$x_4 \leq 3100$ for the capacity bound of S$_4$

$x_i \geq 0, \quad i = 1,2,3,4,$

$h = 1.995;$ **(qualitative evaluation for the crisp number)**

$h = 0.995;$ **(qualitative evaluation for the unconvinced decision)**

$h \leq 0.59;$

$h \geq 0.475;$ **(qualitative evaluation for the in between)**

$d_1^+, d_1^-, d_2^+, d_2^-, d_3^+, d_3^-, d_4^+, d_4^-, e_1^+, e_1^-, e_2^+, e_2^- \geq 0.$

These models can be solved using LINGO [19] to obtain optimal solutions. The best suppliers and their optimum quantities are calculated as follows: For the crisp number results: $S_1(x_1=2700)$, $S_3(x_3=907)$, $S_2(x_2=0)$, and $S_4(x_4=0)$ with TVP= 13711.56. For the unconvinced decisions' results: $S_1(x_1=2700)$, $S_3(x_3=907)$, $S_2(x_2=0)$, and $S_4(x_4=0)$ with TVP=13711.56. For the in between results: $S_1(x_1=2700)$, $S_3(x_3=907)$, $S_2(x_2=0)$, and $S_4(x_4=0)$ with TVP=8608.82. The results are consistent with the Liao and Kao [18]'s results.

5 Conclusion

Due to the difficultness in determining the value of some parameters because of their uncertain or ambiguous nature, this dissertation has developed a new qualitative evaluation that considered three different aspects which are linguistic to crisp, the unconvinced decision and in between. This new qualitative evaluation was developed to produce an optimal preference ranking of the integrated fuzzy TOPSIS and MCGP in interval type-2 fuzzy sets (IT2 FSs) aspects. An example was used and cited without modification to illustrate the procedures and feasibility of the proposed integrated IT2 FTOPSIS in MCGP framework. Five criteria and four alternatives based on one proposed method has been account in this study. The efficiency of using this new method is proven with a straight forward computation in illustrative examples. This approach is successfully provided the qualitative evaluation to the proposed method. Besides, it seems to provide a new perspective in type-2 decision making area. They offer a practical, effective and simple way to produce a comprehensive judgment. This research can further be extended by using non-symmetrical interval triangular and trapezoidal T2 FS.

Acknowledgement. This research is supported by MyBrain15 scholarship and Fundamental Research Grant, no. 59243. This support is gratefully acknowledged.

References

1. Misra, S.K., Ray, A.: Integrated AHP-TOPSIS Model for Software Selection Under Multi-criteria Perspective. In: Driving the Economy through Innovation and Entrepreneurship, pp. 879–890 (2013)

2. Wittstruck, D., Teuteberg, F.: Integrating the Concept of Sustainability into the Partner Selection Process: A fuzzy-AHP-TOPSIS Approach. International Journal of Logistics Systems and Management 12, 195–226 (2012)
3. Mofarrah, A., Husain, T., Hawboldt, K.: Decision Making for Produced Water Management: An Integrated Multi-Criteria Approach. International Journal of Environmental Technology and Management 16, 102–128 (2013)
4. Haldar, A., Banerjee, D., Ray, A., Ghosh, S.: An Integrated Approach for Supplier Selection. Procedia Engineering 38, 2087–2102 (2012)
5. Hsu, C.-H., Yang, C.-M., Yang, C.-T., Chen, K.-S.: An Integrated Approach for Innovative Product Development and Optimal Manufacturer Selection. International Journal of Information and Management Sciences 24, 107–116 (2013)
6. Mirhedayatian, S.M., Vahdat, S.E., Jelodar, M.J., Saen, R.F.: Welding Process Selection for Repairing Nodular Cast Iron Engine Block by Integrated Fuzzy Data Envelopment Analysis and TOPSIS Approaches. Materials and Design 43, 272–282 (2013)
7. Kamis, N.H., Daud, M., Sulaiman, N.H., Abdullah, K., Ibrahim, I.: An Integrated Fuzzy Approach to Solving Multi-Criteria Decision Making Problems. In: IEEE Symposium on Humanities, Science and Engineering Research (SHUSER), pp. 1591–1596 (2012)
8. Nakhaeinejad, M., Nahavandi, N.: An Interactive Algorithm for Multi-Objective Flow Shop Scheduling with Fuzzy Processing Time through Resolution Method and TOPSIS. International Journal of Advanced Manufacturing Technology 66, 1047–1064 (2013)
9. Shidpour, H., Shahrokhi, M., Bernard, A.: A multi-objective programming approach, integrated into the TOPSIS method, in order to optimize product design; In three-dimensional concurrent engineering. Computers and Industrial Engineering 64, 875–885 (2013)
10. Jozi, S.A., Shafiee, M., Moradi Majd, N., Saffarian, S.: An Integrated Shannon's Entropy–TOPSIS Methodology for Environmental Risk Assessment of Helleh Protected Area in Iran. Environmental Monitoring and Assessment 184, 6913–6922 (2012)
11. Iç, Y.T.: An Experimental Design Approach using TOPSIS Method for the Selection of Computer-Integrated Manufacturing Technologies. Robotics and Computer-Integrated Manufacturing 28, 245–256 (2012)
12. Khalili-Damghani, K., Tavana, M., Sadi-Nezhad, S.: An Integrated Multi-Objective Framework for Solving Multi-Period Project Selection Problems. Applied Mathematics and Computation 219, 3122–3138 (2012)
13. Yu, V.F., Hu, K.-J.: An Integrated Fuzzy Multi-Criteria Approach for the Performance Evaluation of Multiple Manufacturing Plants. Computers and Industrial Engineering 58, 269–277 (2010)
14. Chen, S.-M., Lee, L.-W.: Fuzzy Multiple Attributes Group Decision-Making based on the Ranking Values and the Arithmetic Operations of Interval Type-2 Fuzzy Sets. Expert Systems with Applications 37, 824–833 (2010)
15. Chen, S.-M., Lee, L.-W.: Fuzzy Multiple Attributes Group Decision-Making based on the Interval Type-2 TOPSIS Method. Journal of Expert Systems with Application 37, 2790–2798 (2010)
16. Celik, E., Bilisik, O.N., Erdogan, M., Gumus, A.T., Baracli, H.: An Integrated Novel Interval Type-2 Fuzzy MCDM Method to Improve Customer Satisfaction in Public Transportation for Istanbul. Transportation Research Part E 58, 28–51 (2013)
17. Liou, T.S., Wang, M.T.: Ranking Fuzzy Numbers with Integral Value. Fuzzy Sets and Systems 50, 247–255 (1992)
18. Liao, C.-N., Kao, H.-P.: An Integrated Fuzzy TOPSIS and MCGP Approach to Supplier Selection in Supply Chain Management. Expert Systems with Applications 38, 10803–10811 (2011)
19. Schrage, L.: LINGO Release 8.0. LINDO System, Inc. (2002)

A Performance Comparison of Genetic Algorithm's Mutation Operators in *n*-Cities Open Loop Travelling Salesman Problem

Hock Hung Chieng and Noorhaniza Wahid

Faculty of Computer Science and Information Technology,
University Tun Hussein Onn Malaysia,
86400 Parit Raja, Batu Pahat, Johor, Malaysia
joshua_89chieng@hotmail.com,
nhaniza@uthm.edu.my

Abstract. Travelling Salesman Problem (TSP) is one of the most commonly studied optimization problem. In Open Loop Travelling Salesman Problem (OTSP), the salesman travels to all the given *m* cities but does not return to the city he started and each city is visited by salesman exactly once. However, a new problem of OTSP occur when the salesman does not visit all the given *m* cities, but only to visit *n* cities from the given *m* cities. This problem called *n*-Cities Open Loop Travelling Salesman Problem (*n*OTSP), which seems to be more close to the real-life transportation problem. In this paper, Genetic Algorithm (GA) with different mutation operators is implemented to the *n*OTSP in order to investigate which mutation operators give the optimal solution in minimizing the distance and computational time of the *n* visited cities. The mutation operators are inversion, displacement, pairwise swap and the combination of the above three operators. The results of these comparisons show that the GA-inversion mutation operator can achieve better solution in minimizing the total distance of the tour. In addition, the GA with combination of three mutation operators has great potential in reducing the computation time.

Keywords: Genetic Algorithm, *n*-Cities Open Loop Travelling Salesman Problem (*n*OTSP), mutation operators.

1 Introduction

TSP is one of the NP-hard problems that have been widely studied in combinatorial optimization field. It was first introduced by two mathematicians in 1800s [1]. Later, the problem was first formulated by Karl Menger in 1930 [2]. In TSP, a salesman and a list of cities were given. The salesman was supposed to travel to visit all the given cities to sell his goods and return to the city that he started from. The goal is to visit all the cities and return to the starting point with minimum total distance.

Although the origin of the TSP was devoted to a complete closed Hamilton path which means a path that visits every nodes in the graph exactly one and returning to

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_9, © Springer International Publishing Switzerland 2014

the starting point, the research on Open Loop Travelling Salesman (OTSP) Problem is still limited. Nevertheless, in the real-life transportation problem, the salesman does not need to visit all the given *m* cities, but only to visit *n* cities from the given *m* cities. This problem is called *n*-Cities Open Loop Travelling Salesman Problem (*n*OTSP). This problem can be applied in some of the route planning scenarios. For example, logistic transportation routing of merchandise delivery, the delivery start from the depot to the destination without passing through all the cities but only limited number of cities is required.

Fig. 1 shows the difference between OTSP and *n*OTSP. Salesman in Fig.1 (a) departs from a starting point to a target point by visiting all the given *m* cities only once with minimum total distance. However, the salesman in Fig.1 (b) departs from a starting point to a target point and not visiting all the given *m* cities, but he is constrained to visit only *n* cities with minimum total distance. To the best of our knowledge, there are only two researches regarding OTSP can be found. Vashisht [3] have implemented Genetic Algorithm (GA) in OTSP, and seems that GA has proved its suitability to the OTSP. But somehow GA is facing with difficulty in maintaining the optimal solution over many generations. Wang [4] has proposed a Simple Model (SModel) to be implemented in TSP with multi depots and open paths (mdop) to determine the best number of salesman with nearly minimum total distance.

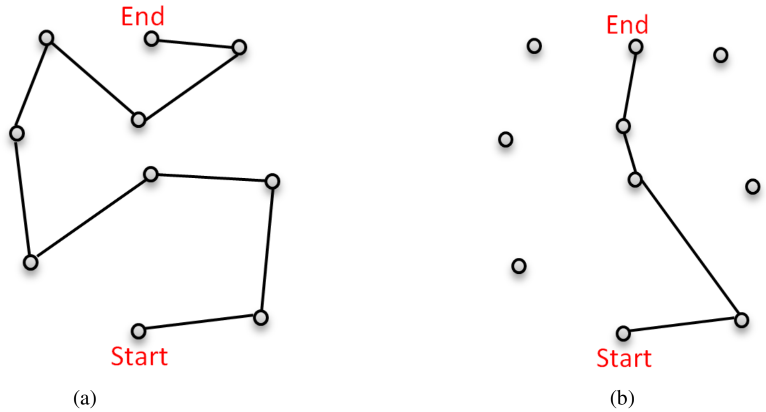


Fig. 1. Two types of Open Loop Travelling Salesman Problem. (a) Classic Open Loop Travelling Salesman Problem (OTSP), (b) *n*-Cities Open Loop Travelling Salesman Problem (*n*OTSP).

In the past, many of the meta-heuristic algorithms have been applied in optimization problems. Until this moment, there are some algorithms that were applied in this problem, such as Tabu Search (TS) [5], Particle Swarm Optimization (PSO) [6], Ant Colony Optimization (ACO) [7] [8], Genetic Algorithm (GA) [9], and Simulated Annealing (SA) [10]. Among the meta-heuristic algorithms, GA has been highlighted to have good performance in solving many combinatorial optimizations [11]. The successfulness of the algorithm depends on its operators such as selection, crossover and mutation operators [12]. Therefore, various methods based on GA were developed to solve the TSP.

This paper presents an extension of OTSP named *n*-Cities Open Loop Travelling Salesman Problem (*n*OTSP) with different GA mutation operators (inversion mutation, displacement mutation, pairwise swap mutation and the combination of these three mutation operators) are implemented to investigate the effectiveness and efficiency of the operators in minimizing the total distance and reducing the computational time. In the experiment, a set of cities *m* and four sets of constrained visited city *n* are given. Each set of *n* is tested using GA with different mutation operators using MATLAB R2010a.

The rest of the paper is arranged as follows: The following section will formally give an overview of the GA and its operators. In Section 3 brief description of the three different mutation operators that used during the experiment. Section 4 presents the experimental result and some analysis. Finally, the conclusion and direction of future work are given in Section 5.

2 Genetic Algorithms: An Overview

Genetic Algorithms (GAs) [13] is the most popular technique in evolutionary computation research. It was purposed based on the survival of the fittest idea in 1975 by Holland. In its nature, GAs operate by come out with sets of solutions called "populations" or usually referred to as "chromosomes" [14]. Chromosomes are the place whereby all the genetic information is stored [13]. Each chromosome is formed by "gene" and it determines the solution [15]. In artificial intelligent, GAs start working by randomly generating a set of solutions (or population). After that, the solutions are evaluated to determine the fittest individuals (parents) for a mating process in order to produce new set of solutions. This process continues for many generations until the condition is met.

2.1 Genetics Operators

A genetic operator used in GA is to maintain the genetic diversity and to combine existing solutions into other. GA that has four important operators applied onto the individuals is explained below.

2.1.1 Encoding: Encoding operator transforms the problem solution into chromosome or called the gene sequence. Encoding techniques such as binary encoding, permutation encoding, value encoding and tree encoding can be applied according the model of the problem [16].

2.1.2 Selection: The selection operator selects members from its population based on their fitness to enter a mating pool [16] [15]. Those individuals nearer to the solution have high chance to be selected. There are many selection methods like roulette wheel selection, proportional selection, ranking selection, tournament selection, range selection, gender-Specific selection (GACD) and GR based selection [18].

2.1.3 **Crossover:** Crossover is the recombining (mating) process of two chromosomes (parents) and producing from them a child (new chromosome or offspring) [15]. Crossover operator is applied to the parents to expect the better offspring is produced. Crossover techniques are single point crossover, two point crossover, multi-point crossover, uniform crossover, three parent crossover, etc. [15][17].

2.1.4 **Mutation:** Mutation is performed after crossover. The purpose of mutation is to prevent the algorithm from being trapped in a local minimum and increase the genetic diversity in the population [14]. Many types of Mutation techniques such as flipping mutation, interchanging mutation, boundary mutation and reversing mutation.

3 Mutation Operators

Mutation operator has very significant role in Genetic Algorithm. Fogel [19][20][21][22] has claimed that only by mutation itself can do everything and it is very useful for function optimization task. In this study, mutation operators used to implement in *n*-Cities Open Loop Travelling Salesman Problem (*n*OTSP) are presented in this section. A set of string (1 5 9 3 7 4 6 2 8 0) represents a tour with 10 cities. It means the salesman will go from city 1 to city 5 to city 9 to city 3 and stop at city 0. Before a selected population is implemented by mutation operator, two cities are randomly chosen from the given string. Let's assume that the city 9 and city 2 are chosen in this case.

3.1 Inversion Mutation

The inversion mutation performs inversion of substring between two selected cities. Fig.2 explains the inversion mutation concept. Two selected cities are city 9 and city 2 then the substring which is **(9 3 7 4 6 2).** After the inversion mutation is performed, the substring **(9 3 7 4 6 2)** will be inverted and become **(2 6 4 7 3 9).**

Before	1 5 **9 3 7 4 6 2** 8 0
After	1 5 **2 6 4 7 3 9** 8 0

Fig. 2. Before and after the inversion mutation is performed

3.2 Displacement Mutation

Displacement mutation pulls the first selected gene out of the set of string and reinserts it into a different place then sliding the substring down to form a new set of string. The city 9 is taken out from the tour and placed behind of the city 2 then at the same time the substring **(3 7 4 6 2)** is slide down to fill the empty space. This is shown in Fig. 3.

Before	1 5 **9 3 7 4 6 2** 8 0
After	1 5 **3 7 4 6 2 9** 8 0

Fig. 3. Before and after the displacement mutation is performed

3.3 Pairwise Swap Mutation

For pairwise swap mutation the residues at the two positions chosen at random are swapped. Sometimes this technique is called interchange mutation or random swap [14]. During the operation, the location of city 9 and city 2 will be swapped. This is shown in Fig. 4.

Before	1 5 **9** 3 7 4 6 **2** 8 0
After	1 5 **2** 3 7 4 6 **9** 8 0

Fig. 4. Before and after the pairwise swap mutation is performed

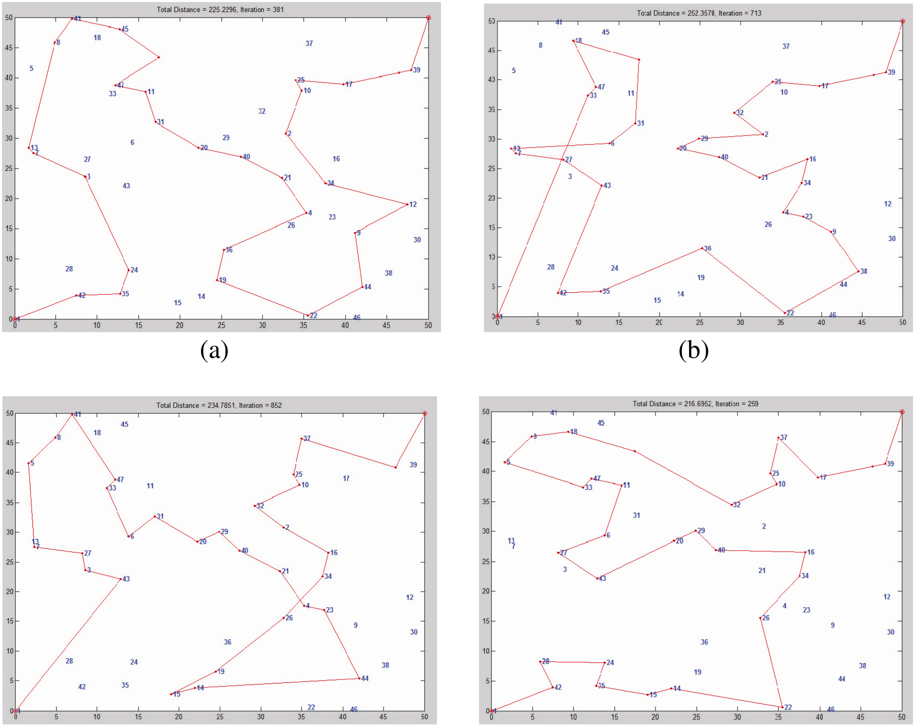
4 Experimental Result

We implemented all of GA with different mutation operator in MATLAB R2010a on a PC machine with Core i5 2.30GHz in CPU and 8GB of RAM with a Window 7 as an operating system. For statistically comparison, each GA with different mutation operator is executed 10 times on *n*OTSP. In this case, population of 1000 chromosomes is used, iteration is set 1000, total number of cities *m* is 50 and number of constrained visited city *n* are 10, 20, 30 and 40.

The experiment focuses on comparing the mutation operators to see which mutation operator gives the best result for finding the minimum total distance and the shortest computation time for generating its optimal solution. The computational result of *n*OTSP with *n*=30 using GA-inversion mutation, GA-displacement mutation, GA-pairwise swap mutation and GA-inversion, displacement and pairwise swap mutation are depicted in Fig. 5(a), Fig. 5(b), Fig. 5(c) and Fig. 5(d), respectively. The numbers represent the visited cities and red line represents the path of the tour.

From the figures above, Fig. 5(b) and Fig. 5(c) clearly show that the crossing path produced by GA with displacement and GA with pairwise swap as its mutation operators. Relatively, the path in Fig. 5(a) and Fig. 5(d) are non-crossing path which produced by GA-inversion mutation and GA-inversion, displacement and pairwise swap mutation.

Table 1 presents the average computation result over 10 runs using GA with different mutation operators in four different number of constrained visited city *n* (*n*=10, 20, 30 and 40). The column shaded in green color with bold numerical value indicates the best average result. Table 1 contains three sections which are Section A,



(a) (b)

(c) (d)

Fig. 5. Computational result of *n*OTSP with *n*=30. (a) GA-inversion mutation, (b) GA-displacement mutation, (c) GA-pairwise swap mutation, (d) GA-inversion, displacement and pairwise swap mutation.

Section B and Section C. In Section A, the values indicate the average minimum distance for different mutation operators with four different number of *n* (*n*=10, 20, 30 and 40).

From the result, GA with inversion mutation operator seems to be outstanding in generating the minimum distance for all four different number of constrained visited city *n*. Relatively, GA with displacement mutation operator and GA with pairwise swap mutation operator did not produce a good solution, because they tend to produce the paths with crossing each other as shown in Fig. 5(b) and Fig. 5(c). Essentially, a path with minimum total distance should not have two lines crossing each other, since that the total distance of crossing path is greater than the non-crossing path [23][24]. Likewise, the results also show that GA with combination of the three mutation operators has not given better solution due to the drawback (crossing paths) of the displacement mutation operator and pairwise swap mutation operator. At the same time, the drawback has affected the entire computational process and solutions.

Table 1. Average over 10 runs using GA with different mutation operator for problems of four different number of constrained visited city *n* (*n*=10, 20, 30 and 40)

Number of constrained visited city (*n*)	**Mutation operators**			
	Inversion	**Displacement**	**Pairwise Swap**	**Inversion + Displacement + Pairwise Swap**
	Section A: Average of minimum tour distance			
10	**105.1791303**	109.6361094	112.1044998	116.8613086
20	**174.3624949**	178.0269224	186.5919048	184.9301064
30	**222.1828426**	237.0308001	254.4681519	228.5520741
40	**255.7966691**	291.6375579	318.0730783	261.728686
	Section B: Average iteration			
10	32.6	31.5	42.4	**16.5**
20	180.8	346.9	341.6	**139.6**
30	564.6	641.1	720.8	**276.4**
40	884.3	933.6	938.8	**420.3**
	Section C: Average of computation time (*s*)			
10	10.4	11.2	10.3	**4.1**
20	11.2	12.2	11.5	**5**
30	14.1	14.1	13.7	**6**
40	15.8	15.8	15.6	**7.1**

Besides that, in Table 1, Section B and C show that the GA with a combination of three mutation operators contribute lesser iteration and computation time. This is due to the capability of each mutation operator that contributes its optimal solutions as well as increasing the probability in searching the optimal solution,

Somehow the finding is seemed to have contradiction with the finding in [25]. The experiment in [25] had applied GA with different mutation operators in classic TSP showed that GA with displacement mutation operator outperformed GA with inversion mutation operator. There are some circumstances should be noticed such as the behavior of the problems (*n*OTSP is differ from TSP), implementation language and environment (such as Maltab, Borland Delphi, JAVA and C++) and the structure of the algorithm. All these can probably be the causes that influence the computational results and findings.

5 Conclusion

This paper investigates the performance of GA with different mutation operators in a new TSP problem called *n*-Cities Open Loop Travelling Salesman Problem. Result shows that GA using inversion mutation operator gave better results in minimizing the total distance. In additional, the experimental results also show that GA with combination of three mutation operators (inversion, displacement and pairwise swap mutation operator) shows to have great potential in minimizing the computation time. Generating good solution for the TSP using GA is still depend on how the problem is encoded and the right operator is applied. According to the comparative study of the mutation operators mentioned, the development of innovation mutation operators for the travelling salesman problem may be the subject of the future research.

For the next step, we are also planning to implement GA with Greedy Sub Tour Mutation (GA-GSTM) [25] in *n*OTSP. According to [25] GA-GSTM has been implemented in classic TSP showed to have better result compared to GA with other mutation operators. Additionally, Simplified Swarm Optimization (SSO) [26] are also one of the new algorithm that can be explored, due to the algorithm having a better performance in finding good and accurate solution in [26]. Hence, the result can also be taken to compare with GA to see which algorithm has a better performance.

Acknowledgment. This work was supported by the Office for Research, Innovation, Commercialization and Consultancy Management (ORICC) of University Tun Hussein Onn Malaysia, Johor. Grant funded by the Graduate Incentive Scheme Vot 1258.

References

1. Matai, R., Singh, S., Mittal, M.L.: Traveling Salesman Problem: an Overview of Applications, Formulations, and Solution Approaches. Traveling Salesman Problem, Theory and Applications. InTech (2010)
2. Maredia, A.: History, Analysis, and Implementation of Traveling Salesman Problem (TSP) and Related Problems. Doctoral dissertation, University of Houston (2010)
3. Vashisht, V.: Open Loop Travelling Salesman Problem using Genetic Algorithm. International Journal of Innovative Research in Computer and Communication Engineering 1(1) (2013)
4. Wang, X., Liu, D., Hou, M.: A novel method for multiple depot and open paths, Multiple Traveling Salesmen Problem. In: IEEE 11th International Symposium on Applied Machine Intelligence and Informatics (SAMI), pp. 187–192 (2013)
5. Basu, S.: Tabu Search Implementation on Traveling Salesman Problem and Its Variations: A Literature Survey. American Journal of Operations Research 2(2), 163–173 (2012)
6. Yan, X., Zhang, C., Luo, W., Li, W., Chen, W., Liu, H.: Solve Traveling Salesman Problem Using Particle Swarm Optimization Algorithm. International Journal of Computer Science, 264–271 (2012)
7. Junjie, P., Dingwei, W.: An ant colony optimization algorithm for multiple travelling salesman problems. In: First International Conference on Innovative Computing, Information and Control, vol. 1, pp. 210–213. IEEE (2006)
8. Bai, J., Yang, G.K., Chen, Y.W., Hu, L.S., Pan, C.C.: A model induced max-min ant colony optimization for asymmetric traveling salesman problem. Applied Soft Computing 13(3), 1365–1375 (2013)
9. Liu, F., Zeng, G.: Study of genetic algorithm with reinforcement learning to solve the TSP. Expert Systems with Applications 36(3), 6995–7001 (2009)

10. Wang, Y., Tian, D., Li, Y.H.: An Improved Simulated Annealing Algorithm for Travelling Salesman Problem. In: Lu, W., Cai, G., Liu, W., Xing, W. (eds.) Proceedings of the 2012 International Conference on Information Technology and Software Engineering. LNEE, vol. 211, pp. 525–532. Springer, Heidelberg (2013)
11. Bahaabadi, M.R., Mohaymany, A.S., Babaei, M.: An Efficient crossover operator for travelling salesman. International Journal of Optimization in Civil Engineering 2(4), 607–619 (2012)
12. Abdoun, O., Abouchabaka, J.: A Comparative Study of Adaptive Crossover Operators for Genetic Algorithms to Resolve the Traveling Salesman Problem. arXiv preprint arXiv:1203.3097 (2012)
13. Sivanandam, S.N., Deepa, S.N.: Introduction to genetic algorithms. Springer (2007)
14. Sallabi, O.M., El-Haddad, Y.: An Improved Genetic Algorithm to Solve the Traveling Salesman Problem. World Academy of Science, Engineering and Technology 52, 471–474 (2009)
15. Geetha, R.R., Bouvanasilan, N., Seenuvasan, V.: A perspective view on Travelling Salesman Problem using genetic algorithm. In: Nature & Biologically Inspired Computing. World Congress, pp. 356–361 (2009)
16. Malhotra, R., Singh, N., Singh, Y.: Genetic algorithms: Concepts, design for optimization of process controllers. Computer and Information Science 4(2), 39–54 (2011)
17. Geetha, R.R., Bouvanasilan, N., Seenuvasan, V.: A perspective view on Travelling Salesman Problem using genetic algorithm. In: Nature & Biologically Inspired Computing. World Congress, pp. 356–361. IEEE (2009)
18. Sivaraj, R., Ravichandran, T.: A review of selection methods in genetic algorithm. International Journal of Engineering Science and Technology (IJEST) 3(5), 3792–3797 (2011)
19. Fogel, L.J., Owens, A.J., Walsh, M.J.: Artificial intelligence through simulated evolution (1966)
20. Fogel, D.B.: Empirical estimation of the computation required to reach approximate solutions to the travelling salesman problem using evolutionary programming, vol. 685, pp. 56–61 (1993)
21. Fogel, D.B., Atmar, J.W.: Comparing genetic operators with Gaussian mutations in simulated evolutionary processes using linear systems. Biological Cybernetics 63(2), 111–114 (1990)
22. Fogel, D.B.: Evolutionary computation: toward a new philosophy of machine intelligence, vol. 1. John Wiley & Sons (2006)
23. Yong, S., Zenglu, L., Wenwei, L., Zhongkai, Y., Guangyun, L., Jirong, X.: The research and application on improved intelligence optimization algorithm based on knowledge base. In: 2012 International Conference on Computer Science and Electronics Engineering (ICCSEE), vol. 3, pp. 661–665. IEEE (2012)
24. Applegate, D.L., Bixby, R.E., Chvatal, V., Cook, W.J.: The traveling salesman problem: a computational study. Princeton University Press (2011)
25. Albayrak, M., Allahverdi, N.: Development a new mutation operator to solve the Traveling Salesman Problem by aid of Genetic Algorithms. Expert Systems with Applications 38(3), 1313–1320 (2011)
26. Chung, Y., Bae, C., Wahid, N., Liu, Y., Yeh, W.C.: A New Simplified Swarm Optimization (SSO) Using Exchange Local Search Scheme. International Journal of Innovative Computing, Information and Control 8(6), 4391–4406 (2012)

A Practical Weather Forecasting for Air Traffic Control System Using Fuzzy Hierarchical Technique

Azizul Azhar Ramli, Mohammad Rabiul Islam, Mohd Farhan Md Fudzee,
Mohamad Aizi Salamat, and Shahreen Kasim

Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia, Parit Raja, 86400 Batu Pahat,
Johor Darul Takzim, Malaysia
{azizulr,farhan,aizi,shahreen}@uthm.edu.my, rabisp@yahoo.com

Abstract. Due to rapid changes of global climate, weather forecasting has becomes one of the significant research fields. Modern airports maintain high security flight operations through precise knowledge of weather forecasting. The objectives of this research focused on two major parts; the weather forecasting model of an airport system and the fuzzy hierarchical technique used. In general, this research emphasizes on the building blocks of a weather forecasting application that could support Terminal Aerodrome Forecast by utilizing Mamdani model. The developed application considers variables, groups of weather elements, combination of weather elements in a group, web data sources and structured knowledge to provide a profound forecast.

Keywords: Fuzzy Logic, Weather Forecasting, Air-Traffic Control, Hierarchical Model.

1 Introduction

The weather forecasting becomes one of the prominent challenges of the 21th century. It is reflected by environmental pollution, global warming and man-made infrastructure and the meteorological changed. This research is mainly concern with solution based on fuzzy logic technique that involved with natural phenomena like ambiguous, imprecise and continuous variables that falls under the theme of natural problem-solving techniques [1]. Meteorological problem is commonly solved through fuzzy logic, neuro-fuzzy predictor and neural network [2].

In this research, fuzzy logic is applied on weather forecasting due to a predetermined condition of weather as the airport regular routine task. The developed fuzzy solution would be beneficial for Engineers or Air Traffic Controllers for the capability of instant weather prediction. In this technique, hierarchical model assists to produce the accurate weather prediction with the mental workload in the fuzzy application system before the procedural of final output of airport weather prediction. The application's prediction strategy is based on instant weather condition of an airport thus, the air-traffic controller may produce the final output either suitable or

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_10, © Springer International Publishing Switzerland 2014

unsuitable condition of flight operation. According to airport system criteria, the safety of the airport depends on instant weather condition based on every six hours weather forecasting [3].

The objective of the paper is to develop a weather forecasting application for airport flight operation, by integrating the hierarchical model with fuzzy logic. The aim is to get the accurate forecast from the conditional impact of meteorological data for flight operation in an airport.

2 Related Work

Through this section, related works which become the foundation towards this research have been comprehensively described.

2.1 Multi-input Criteria in Hierarchical Model

The mechanism of hierarchical Fuzzy Logic Controller (FLC) constructed on the rule-based system in which a set of fuzzy rules represents a control decision mechanism to adjust the effects of certain system disturbances.

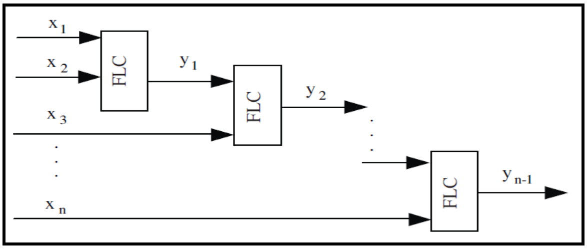


Fig. 1. Hierarchical FLC Structure Representation [4]

In this hierarchical model, the first level gives an approximate output which is then modified by the second-level rule set that shown in Fig. 1. The procedure can be repeated until the last levels of hierarchy [4]. The application of the Hierarchical Fuzzy superiority has been proved for Multi Input System (HFMSS).

2.2 Fuzzy Logic Technique Applied on Weather Forecasting

Weather forecasting used in different sectors is based on level of importunacy. It is observed that, nearly 25 percents of fatal weather-related accidents involved thunderstorms [5]. Air Traffic Controller (ATC) encountered the effects of a daily weather condition in terms of heavy raining, temperature and barometric pressure, thunderstorms, drizzling and haze. Fuzzy weather forecasting model determines the instant weather forecast by the Air-Traffic Controller (ATCo) to control flight. Typically, weather problems are interpreted through the fuzzy mechanism to enable the extendable solution. Next, a detailed explained of each problem is presented.

i. *Visibility of Vision Blurs*
   Visibility directly involves with different types of weather elements. Haze, fog, sand or dust storms, cloudy sky, industrial gas or chemical reaction through natural air cause the blur of pilot visibility which is not indispensable for landing and departing operation for an airport.
ii. *Wind Speed of Natural Calamities*
   Wind caused the natural devastation due to its speeds and direction of place. In this project, wind speed measurement is provided through fuzzy technique instead of indication of wind direction to get the accurate forecast.
iii. *Airport Turbulence*
   Turbulence is considered as flight obstacle of weather that usually happened during departing and landing of a plane. Temperature, dew-point and relative humidity are applicable as an integrated element in fuzzy logic turbulence form. In this research, we identified the problem as the limitation of proper information about climate information between pilot and ATCo; they need a solid understanding of climate condition in order to make sure that the conditional result not only depends on radar services but also the current situation. This is made possible by the fuzzy weather forecasting model that can handle more than two inputs simultaneously or hierarchically to support defuzzification and produce the single output as the informed result.

2.3 Fuzzy Logic Configuration Model in Systematic Operation of Airport

Human Traffic Flow Management (TFM) expertise and machine based decision support system is conveying effective air traffic capacity [6]. The Next Generation Air Transportation System (NGATS) is free of traffic growth whether Air Traffic Management (ATM) capacity demands will be imbalanced that may help verities operations to resolve with techniques [6]. Usually such operation could be solved by Consolidated Storm Prediction for Aviation (CoSPA) step in ATM System. The CoSPA configuration is given below in Fig. 2.

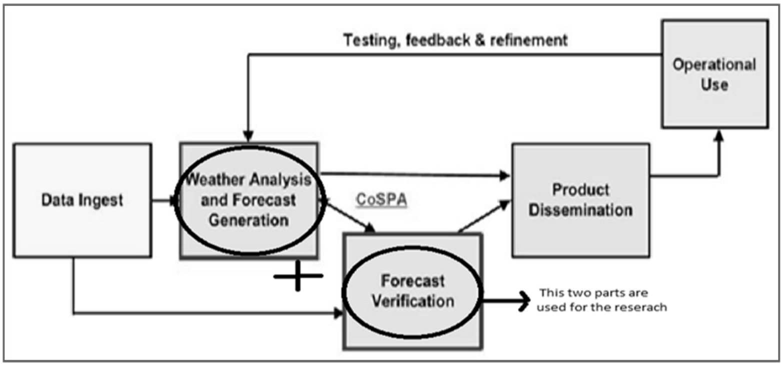

Testing, feedback & refinement
Operational Use
Data Ingest
Weather Analysis and Forecast Generation
CoSPA
Product Dissemination
+
Forecast Verification
This two parts are used for the reserach

Fig. 2. Developing an aviation oriented forecast must involve feedback between the experimental operational use of the forecast and refinement of the forecast [7]

ATM community efficiently depends on scenarios of increased demand of airport usage. Automated decision support tools help to make usable probabilistic weather guidance that estimate airspace capacity for managing air traffic flows [8]. The proposed solution is developed based on ATC that must address the user communication requirements of the mature ATM system that provide various data links [9]. The fuzzy logic complex mechanism helps to complete the flexibility of forecasting operation.

3 Application Techniques of General Hierarchical Model

Hierarchical model is a design structure that consists of multiple input levels with specific components among different values or elements [10]. Hierarchical method is performed by different functionality of live features and potential evaluation to produce the final result [11]. Typically, the hierarchical model is used in many research fields to support different ways of achieving any research goal. Examples are: hierarchical clustering on neural network [12], relation task hierarchies [13], decision task from a medical hierarchical model [14], and forecasting methodology through Bayesian hierarchical model [15]. The main benefit of the hierarchical model is it allows learning the environment based on modular fashion. Therefore, people are able to learn faster or even able to discover relationships for each step of the abstract level [16].

Similarly, our solution adopted the hierarchical procedure. Figure 3 represents the three combined groups of three elements of weather. Each of the group generates intermediate output (IO), where the final one is based on these outputs.

This model also known as Interactive Systematic Hierarchical (ISH) that has been used in many operations through Research and Development (R&D) [6]. Any systematic analysis of interactions levels and levels of human awareness could be incorporated in the decision through the hierarchical model as experts are always necessary for weather forecasting in musicale in terms of stratus, fog and convection [17].

ISH model was developed to reflect the weather elements that can simultaneously affect and interrelated to one another [6]. All nine weather elements able to produce the weather forecasting situation of airport and concluded through general problem of overall decision from a fuzzy logic application [7] [18].

Generally, fuzzy logic consists of degree of membership value, which could be accepted as the value of partial truth or false, at a same time [19] [20]. The membership is constructed according to Airmen’s Meteorological Information (AIRMET) which is vital operational task to airmen and Federal Aviation Administration (FAA) [21]. The main task of the AIRMET is to describe aviation forecast based on regions. Airport weather forecasting is updated several times a day [20] based on updated weather data from online.

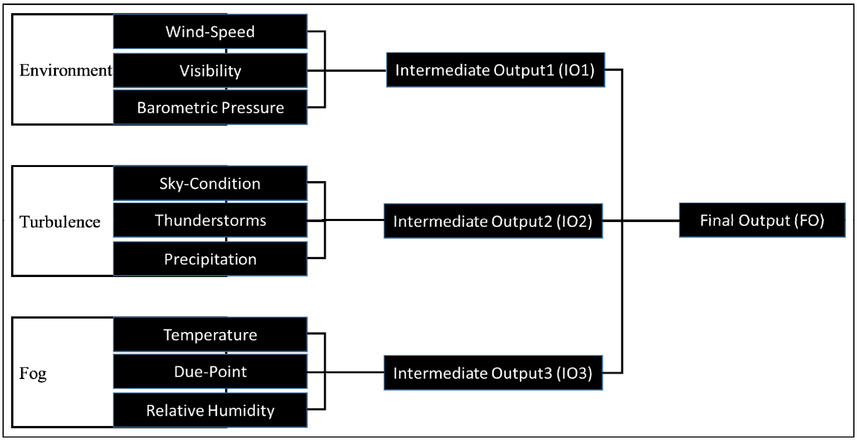


Fig. 3. General hierarchical model for weather combination

Our solution used date from up-to-date website: www.met.gov.my to get the accurate forecast that is maintained by government staff. Automatic and Terminal Information Services (ATIS) also provided some data from an airport radar system which is available on the weather website. From the operational model, the potential hazards or normal situation will be determined accordingly [3].

3.1 Weather Elements of Fuzzy Application Model

Weather elements are categorized with the context of natural conditions like dew point, spread, rate, wind speed and sky as in Table 1. The qualitative description of the fuzzy set has to be chosen as the inputs from the weather phenomenon.

Table 1. Several Fuzzy Sets Based on Natural Condition [3]

Dew Point	Spread	Rate	Wind	Sky
Dry	Unsaturated	Drying	Too light	Cloudy
Moderate	Saturated	Saturating	Excellent	Clear
Moist	Very saturated		Too Strong	
Very moist				

The value of current dew-point is categorized into one of four fuzzy sets described as dry, moderate, moist and very moist. The value of dew-point is to be characterized in two by manipulating the rate of change of the dew point spread, whether the atmosphere is drying (positive rate) or showing a saturating trend (negative rate). Sky condition is classified into two categories either clear or cloudy. Canadian airport system applied similar fuzzy model to forecast weather as routine task [22].

4 Proposed Architecture

Fuzzy logic model was developed through a workflow model that used to develop any interesting operational field, especially skill optimization objective. Real world problems could be solved by designing of optimization techniques with fuzzy algorithm. The research can be conducted based on its control problem or solution. In our case, the algorithm is developed from the control problem and prediction perspective of a fuzzy system as shown in Fig. 4.

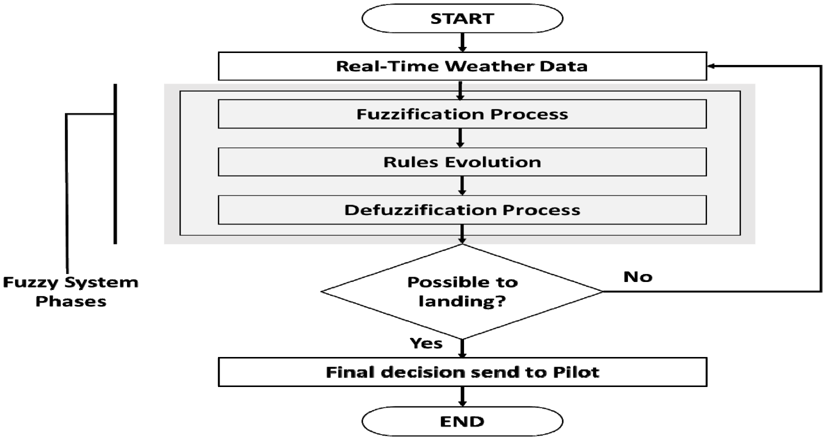


Fig. 4. Weather forecasting fuzzy system basic component

At the starting point of the algorithm, the input to the system used as "crisp" input for the fuzzification of the next step. The third step shows that the fuzzy meteorological data is collected from online weather service. It involves the conversion at the input/output signals into a number of fuzzy represent values. By the step of fuzzy output weather elements, the new conditional fuzzy associative memory table will be created that is also known as FAM table. The next step gets the value of output to execute defuzzification process for "crisp" output of weather as shown in this inference model. Based on the "crisp" output the air traffic controller will decide the condition, whether the airport is suitable for landing or not. If the prediction of weather is verified twitch, then the ATCo begins from the fuzzification step.

4.1 Combination of Different Weather Elements

Earth surface continually generates different pressure. It is important to measure the barometric pressure for air rising with low pressure and air descending with high pressure [3]. It is the main reason to combine the wind speed, dust storm, and humidity thus, measuring the environment in term of fuzzy logic, depicted as Intermediate Output1 (see Figure 3). Besides the wind speed velocity, dust storms caused the obscure of pilot vision, especially during a flight landing or departing on airport operation hours. Both wind velocity and dust storm caused the obstacle of a flight schedule. The combination of three weather inputs gives the prediction on fuzzy logic Intermediate Output1 (*IO1*), Intermediate Output2 (*IO2*) and Intermediate Output3 (*IO3*) as shown in Table 2.

Table 2. Combination of Weather Elements in Hierarchical Model

Categories/Names	**Weather Elements**			**Outputs**
Environment	Wind-Speed	Visibility	Barometric Pressure	Intermediate Output1 (*IO1*)
Turbulence	Sky-Condition	Thunderstroms	Precipitation	Intermediate Output2 (*IO2*)
Fog	Temperature	Due_Point	Relative Humidity	Intermediate Output3 (*IO3*)
				Final Result

According to climatology, the weather condition also applied to turbulence and fog. Sky-condition, thunderstorms and precipitation caused obstacles of airport operation. The first output is environment depicted as *IO1*. The second output is turbulence depicted as *IO2* and third output is fog depicted as *IO3*. The final step is applied from the basic three intermediate outputs (*IO*) of hierarchical model. The first *IO1* grouped with wind speed, visibility and air pressure. The input data are followed as crisp set of weather elements in fuzzy logic MATLAB toolbox. The input data on the fuzzy system considered as crisp numerical values and the output will be set as the fuzzy degree of membership within the qualifying linguistic set. Numbers of fuzzy rules are built that depends on resolving the inputs into a number of different fuzzy linguistic sets. All the input must be fuzzified according to each of these linguistic sets.

Three crisp input values are always set up as a numerical number followed by the universe of discourse. In this case, the hierarchical model structure configured the *IO1* which could be another variable to produce final output. In this model, the *IO1* also prepares the input of fuzzy crisp set. All the steps of weather elements are followed the same procedure of Intermediate Output2 (*IO2*) and Intermediate Output3 (*IO3*).

4.2 Group Based Weather Elements on Fuzzy System

Three groups are combined and described using the MATLAB toolbox operation as below:

i. *Fuzzy Intermediate Output1 (IO1) - Environment*
Fuzzy membership function rule must be determined the natural condition which would be perfect for airport operation.

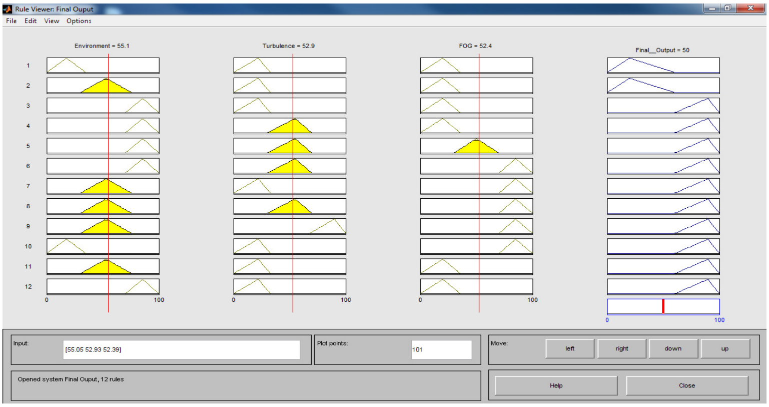
ii. *Fuzzy Intermediate Output2 (IO2) Turbulence*
Clouds, thunderstorms and precipitation are considered for Intermediate Output2. These three weather phenomena are related to each other, and this greatly changes the nature especially in the airport environment and atmosphere.

The combination of weather variables is known as turbulence that can cause interruption of departing and landing at the airport [3]. The combination of clouds, thunderstorms and precipitation are important to measure for main output-2. A thunderstorm is known as a storm that generates lightning and thunder at a same time with heavy clouds. Winds, heavy rain and hail are frequently produced from thunderstorms [23]. Length and thickness cumulonimbus clouds create thunderstorm and growth of that cloud. Heavy precipitation and possibly small hail are also causing the thunderstorm [23]. Turbulence is so dynamic and usually occurs on a very small scale with its difficult phenomenon [24].

iii. *Fuzzy Intermediate Output3 (IO3) - Fog*

Temperature is the most influential weather elements in daily life. Many weather elements could be changed based on temperature condition. For example wind speed, precipitation, humidity, dew-point, storm and many more [23]. So, airport temperature gradient is important to be measured in terms of humidity and dew-point.

Temperature control and other weather elements may directly affect one another, for instance a clear sky are often warmer than cloudy one and clear night usually cooler than cloudy one [23]. Temperature, dew-point and relative humidity values are applied on fuzzy MATLAB according to climatology [25]. This three weather phenomenon measured the fog condition around the airport. Fog or haze caused natural catastrophe in airport operation, especially on flight runway.



Input: [55.05 52.93 52.39]

Plot points: 101

Opened system Final Ouput, 12 rules

Fig. 5. The output result is show the airport is suitable for an airport

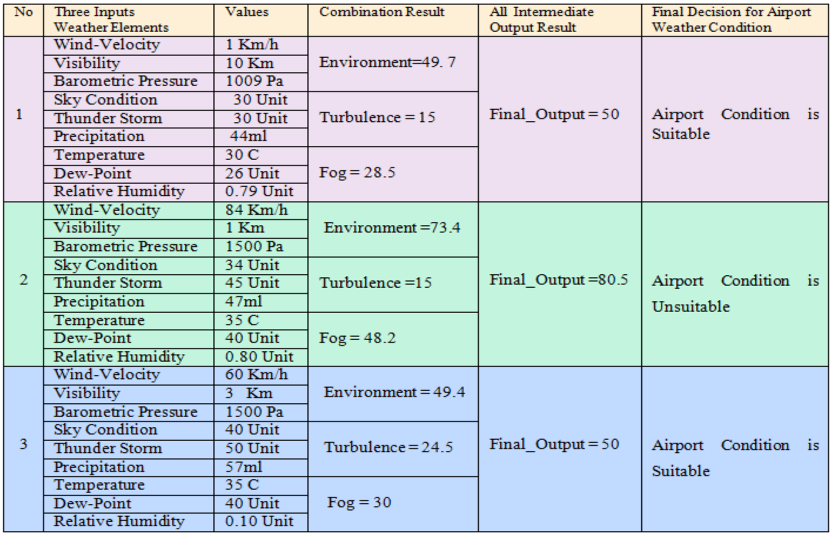
iv. *Final Output of Air-Port Weather Forecasting*

The final output is the fuzzy logic decision and prediction of output as it could be determined from the result of overall weather elements. If one conditional output among these three intermediate outputs (*IO*) gives bad reading, then the final output becomes unsuitable. Figure 5 showed the fuzzy final output from Environment (*IO1*), Turbulence (*IO2*) and Fog (*IO3*).

5 Findings and Discussions

The result and decision from hierarchical method and Mamdani method has shown an accurate result in fuzzy logic application. The specified result and decision making of weather forecasting is produced by executing the application three times with different weather data as shown in Figure 6.

Total of 42 fuzzy inference rules, three classified hierarchical group’s data model and web data source are constructed and used in this fuzzy forecasting application. The main advantage of the application is the ability to determine the condition using the multi values weather elements. Indeed, fuzzy logic can configure and perform this type of operations. Analytical operation of this fuzzy application represents the system reliability, monitoring result and decision making techniques.



No	Three Inputs Weather Elements	Values	Combination Result	All Intermediate Output Result	Final Decision for Airport Weather Condition
1	Wind-Velocity	1 Km/h	Environment=49. 7	Final_Output = 50	Airport Condition is Suitable
	Visibility	10 Km			
	Barometric Pressure	1009 Pa			
	Sky Condition	30 Unit	Turbulence = 15		
	Thunder Storm	30 Unit			
	Precipitation	44ml			
	Temperature	30 C	Fog = 28.5		
	Dew-Point	26 Unit			
	Relative Humidity	0.79 Unit			
2	Wind-Velocity	84 Km/h	Environment =73.4	Final_Output =80.5	Airport Condition is Unsuitable
	Visibility	1 Km			
	Barometric Pressure	1500 Pa			
	Sky Condition	34 Unit	Turbulence =15		
	Thunder Storm	45 Unit			
	Precipitation	47ml			
	Temperature	35 C	Fog = 48.2		
	Dew-Point	40 Unit			
	Relative Humidity	0.80 Unit			
3	Wind-Velocity	60 Km/h	Environment = 49.4	Final_Output = 50	Airport Condition is Suitable
	Visibility	3 Km			
	Barometric Pressure	1500 Pa			
	Sky Condition	40 Unit	Turbulence = 24.5		
	Thunder Storm	50 Unit			
	Precipitation	57ml			
	Temperature	35 C	Fog = 30		
	Dew-Point	40 Unit			
	Relative Humidity	0.10 Unit			

Fig. 6. System test result as a forecasting of the airport condition

The ultimate results of the developed application forward the fuzzy logic technique as a role model in further research fields. Moreover, fuzzy logic technique is extended to a flight schedule at airport system. Fuzzy logic also applicable to improve the WIND-1 mode by testing and prediction of accuracy with systematic airport’s rules, which would be helpful to learn autonomous weather forecasting. Image processing fuzzy application model helps to configure same type of problems [22].

Typical research application model could be applied to the agriculture sector in terms of product, distribution of natural time intensity and many more. In this light, various research application areas are expected to take advantages from this type of fuzzy logic decision-making model.

References

1. Domanska, D., Wojtylak, M.: Fuzzy Weather Forecast in Forecasting Pollution Concentrations. Department of Modeling and Computer Graphics. Institute of Informatics, University of Silesia, Poland (2012)
2. Henson, R.: The Rough Guide to Weather, 2nd edn. Rough Guides Ltd., New York (2007)
3. James, K.: Oxford Aviation Tanning: Ground Training Series. OAT Media, Oxford Aviation Training, Oxford, England (2007)
4. Anarmarz, E.: Hierarchical Fuzzy Controller Applied to Multi-Input Power System Stabilizer. Turki Electric Engineering and Computer Science 18(4) (2010)
5. AOPA Services, AOPA Air Safety Foundation, 2nd edn., Frederick, Maryland, vol. 8(22) (2008)
6. Murtha, J.: Applications of Fuzzy Logic in Operational Meteorology. Scientific Services and Professional Development Newsletter, Canadian Forces Weather Service (1995)
7. Schmidt, R.L., Freeland, J.R.: Recent Progress in Modeling R&D Project Selection Progress. IEEE Transactions on Engineering Management 39(2), 189–200 (1992)
8. Steiner, M.: Translation of Ensemble-Based Weather Forecasts into Probabilistic Air Traffic Capacity Impact, National Center for Atmospheric Research. Matron Aviation, Dulles, VA. IEEE (2009)
9. Gallaher, S.: Communication System Architecture for Air Traffic Management and Weather Information Dissemination. IEEE (2001)
10. Leach, C.W., van Zomeren, M., Zebel, S., Vliek, M.L., Pennekamp, S.F., Doosje, B., Ouwerkerk, J.W., Spears, R.: Group-Level Self-Dimension and Self-Investment: A Hierarchical (Multicomponent) Model of In-Group Identification. Journal of Personality and Social Psychology 95(1), 144–165 (2008)
11. Otavio, A.S.C.: A Hierarchical Hybrid Neural Model in Short Load Forecasting. Instituto de Engenharia Eletrica. IEEE (2000)
12. Jin, L., Feng, Y., Jilai, Y.: Peak Load Forecasting Using Hierarchical Clustering and RPROP Neural Network. Department of Electrical Engineering, Harbin Institute of Technology. IEEE (2006)
13. Natarajan, S.: A Relational Hierarchical Model for Decision-Theoretic Assistance. Oregon State University, Corvallis, Oregon, USA (2008)
14. Moller, K.: Hierarchical Modeling for Medical Decision Support. In: 4th International Conference on Biomedical Engineering and Informatics. IEEE (2011)
15. Werne, J.: Atmospheric Turbulence Forecast for Air Force and Missile Defense Applications. In: DoD High Performance Computing Modernization Program Users Group Conference. IEEE (2010)
16. Theocharous, G.: Learning Hierarchical Partially Observable Markov Decision Process Models for Robot Navigation. In: International Conference on Robotics and Automation. IEEE, Korea (2001)
17. Liao, Z., Greenfield, P., Cheung, M.T.: An Interactive Systematic Hierarchical Model for Strategic R&D Decision Making in a Dynamic Environment. IEEE (1995)
18. Souder, E.W., Mandakovic, T.: R&D Project Selection Models. Research Management 29(4), 36–42 (1986)
19. Michael, N.: Artificial Intelligence: A Guide to Intelligent System, 2nd edn. Addison Wesley (2005)
20. Lopez, R.S.M.: Fuzzy Model for Decision Taking of Technology in Home and Building Electronic Systems. In: Second UKSIM European Symposium on Modeling and Simulation. IEEE (2008)

21. Raymond, L.B.: Airport and Air Traffic Control Advisory Panel Members. NTIS order #PB82-207606 (January 1982)
22. Hansen, B.: A Fuzzy Logic–Based Analog Forecasting System for Ceiling and Visibility. Cloud Physics and Service Weather Research Section, Meteorology Research Division, Environment Canada, Dorval, Quebec, Canada, vol. 22 (December 2007)
23. Lutgens, F.K., Tarbuck, E.J.: The Atmosphere: An Introduction to Meteorology. Prentice Hall, Englewood Cliffs (1995)
24. Tenny, A.L.: Integrated Methods of Diagnosing and Forecasting Aviation Weather. The National Center for Atmospheric Research Boulder. IEEE, Colorado (2000)
25. Peter, M.: Server Flux-Security the Internet. University of Chester (February 2011)

Adapted Bio-inspired Artificial Bee Colony and Differential Evolution for Feature Selection in Biomarker Discovery Analysis

Syarifah Adilah Mohamed Yusoff[1,2], Rosni Abdullah[1], and Ibrahim Venkat[1]

[1] School of Computer Sciences, Universiti Sains, Malaysia
{rosni,ibrahim}@cs.usm.my
[2] Dept. Computer Sciences and Mathematics,
Universiti Teknologi MARA Pulau Pinang, Malaysia
syarifah.adilah@ppinang.uitm.edu.my

Abstract. The ability of proteomics in detecting particular disease in the early stages intrigues researchers, especially analytical researchers, computer scientists and mathematicians. Further, high throughput of proteomics pattern derived from mass spectrometry analysis has embarked new paradigm for biomarker analysis through accessible body fluids such as serum, saliva, and urine. Recently, sophisticated computational techniques that are mimetic natural survival and behaviour of organisms have been widely adopted in problem-solving algorithm. As we put emphasis on feature selection algorithm, the most challenging phase in biomarker analysis is selecting most parsimonious features of voluminous mass spectrometry data. Therefore this study reveals the hybrid artificial bee colony and differential evolution as feature selection techniques exhibits comparable results. These results were compared with other types of bio-inspired algorithms such as ant colony and particle swarm optimisation. The proposed method produced; 1) 100 percent and 98.44 of accuracy of the ovarian cancer dataset; and 2) 100 percent and 94.44 percent for TOX dataset for both training and testing respectively.

Keywords: metaheuristic, feature selection, swarm algorithm, bio-inspired algorithm, classification, ABC, ACO, PSO.

1 Introduction

Mass spectrometry technology is indispensable to bridge the gap between high throughput proteomics and biomarker discovery analysis. Well-known mass spectrometry soft-ionization techniques such as Matrix-Assisted Laser Desorption/ Ionization Time-Of-Flight Mass Spectrometry (MALDI-TOF-MS) and Surface-Enhanced Laser Desorption/Ionization Time-of-Flight Mass Spectrometry (SELDI-TOF-MS) generate high-throughputs of proteomics patterns that are capable in identifying complex protein properties such as peptide sequence and structure of proteins through high-resolution analysis. Common output of

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_11, © Springer International Publishing Switzerland 2014

typical Mass Spectrometry (MS) analysis yields a spectrum from a single sample of data. The spectrum can be represented as a typical xy-graph in terms of ratio of mass to charge ratio (m/z) versus ionization intensities. The peaks of the intensities yield significant information that relates to the biological meaning of the data. Further, the intensity of the peaks characterizes the protein expression level of certain molecules of peptides and leads to discovery new biomarkers for certain disease in different stages. However, this high throughput yields high dimensionality of data where numbers of features are much greater than number of samples.

Feature selection is one of the most promising machines learning technique in defying curse of dimensionality. Recently, bio-inspired metaheuristic algorithms have been widely used for solving combinatorial optimisation problems and yielded promising results [13].

These so-called algorithms have simulated behavior of living things in solving problems and survive in their environment. Given emphasis on feature selection purposes, these algorithms are mainly constructed based on the modern metaheuristic paradigm. This paradigm composes exploration and exploitation behaviour of certain living organisms. Particle Swarm Algorithm (PSO) is one of the most popular bio-inspired algorithms that has been introduced by Kennedy et al. [4] and proposed as feature selection for various problems domain [12]. Along with PSO, Ant Colony Optimisation (ACO) develops as an efficient paradigm of feature selection in the generous domain [1]. Both of these algorithms have been accepted and continuously being adapted in many areas of active ongoing research. Apart from that, Baykasoglu, et al. [3] have reported comprehensive studies of bee systems and demonstrated the potential of bee system as an optimisation technique in complex environment. The beauty of this bee system varies on their characteristic where; (a) marriage behaviour; (b) foraging behaviour; and (c) queen bee concept. One of the promising approaches is Artificial Bee Colony (ABC), which is simple due to its small numbers of parameters despite its capability to solve real world problems.

This algorithm has already been proposed for classification by Mohd Shukran et al. [6] and for feature selection by [13,11] with promising results. However, ABC is still in its infancy and can be improved. Hence, this study intends to improve basic ABC algorithm as feature selection for better convergence and further improve biomarkers discovery in mass spectrometry.

2 Hybrid Artificial Bee Colony and Differential Evolution

Both Artificial Bee Colony and Differential Evolution are two different algorithms that work efficiently for combinatorial optimization problems. ABC reflects the foraging behavior among three different agents; (i) employee bees; (ii) onlooker bees; and (iii) scout bee. Meanwhile DE is the simplest version of algorithm among ES-community that utilized three main concepts of evolution; (a) mutation; (b) crossover; (c) selection.

Figure 1 depicted the whole system of hybrid Artificial Bee Colony and Differential Evolution. The idea of hybridisation is mostly dependent based on how the features are supposed to be manipulated by the algorithms. The parsimonious features can be extracted through single ABC algorithm, though the role of scout bee seems to be less significant in improving abandon features. Instead of relying on current random search, DE has been injected to promote evolution techniques for exploring and exploiting new food sources prior the next search process. Brief discussions of hybridisation ABC and DE are explained as below:

Initialisation of Population: Random population of potential features obtained from feature extraction phase are constructed equally around the search space. Consider the number of employee bees is i and a series of unique solution being optimised by particular bee is d, thus search space is denoted as $X_{i,d}$.

Neighbourhoods Search and Fitness Score: Both employee and onlooker bees will iterate somehow to improve their current nectar amount (fitness score) through exploiting new search space. In this study, exploiting new search space will require neighbourhood search mechanism to randomly modify any $d = 1, 2, 3, , D$, with D as maximum food source for particular X_i. In order to comprehend with the initial data, original expression has been replaced by a simple random search and produced better exploitation results.

$$V_{i,d} = X_{a,b} \tag{1}$$

Equation 1 denoted $V_{i,d}$ as new food source and $X_{a,b}$ is the current solution. Meanwhile, $i = 1, 2, ..., SN$ and $d = 1, 2, ..., D$ respectively, whilst SN is maximum number of employee bees. Apart from that, $X_{a,b}$ must be different from any $V_{i,d}$ for particular i.

Neighbourhood fitness of new food source $V_{i,d}$ is evaluated through minimizing objective function. The objective function is constructed as classification error of linear SVM (Support Vector Machine). The new fitness will be compared with the current fitness of $X_{a,b}$. If new fitness score is superior, then current food source will be replaced and vice versa. The fitness fit_i, of the associated food sources is measured as follow:

$$fit_i = \frac{1}{1 + objVal_i} \tag{2}$$

Probabilistic Selection: Prior onlooker bee selection, fitness of food sources collected by each employee bees will be gathered on the waggle dance area and evaluated as probability score. Further, onlooker bees will select the food source that exhibits the criteria of $rand < P_i$ where;

$$P_i = \frac{fit_i}{(\sum_i^{SN} fit_i)} \tag{3}$$

Abandonment Solution and DE Implementation: Apart of doing neighbourhood improvement to optimise quality nectar amount across the food sources, the exhausted searches also being recorded along the process and controlled by *limit* constant. Exhausted search refers to the agent that cannot perform while neighbourhood exploitation. Nevertheless, the fitness of that exhausted bee is not necessarily the worst fitness among others. Therefore to overcome the exhausted search without directly skipping to other food sources, we injected DE implementation to descent some origin food source. Common implementation has taken only one abandon case to be solved and converted to scout bee. In this study, if *abd* is an array that referred to the worst case of agent bees to be solved, *abd* could be any valued of $1, 2, 3, ., SN$. Since the agents are improved regularly, the maximum exhausted or abandonment bees are average from 3 to 4 in one iteration. DE implementation is described as follow:

- **Mutant.** Abandonment bees for ith iteration is $abd_{i,G}$, where G is number of abandonment bees for each cycle that has exceed *limit* parameter. Thus mutant vector is generated according to equation 4.

  $$v_{i,G} = abd_{r1,G} + \Phi(abd_{r2,G} - abd_{r3,G}) \tag{4}$$

  Where $r1 \neq r2 \neq r3$ are random numbers that will be updated for every single bee, G. Meanwhile $\Phi \in [0, 1]$ is scaling factor that also been updated for every particular i.
- **Crossover.** In general, several mechanisms have been applied for cross over, this study adapts the cross over solution proposed by [10] where lineal and binary recombination have been implemented.Threshold of Cr is set as 0.75 and 0.95 in order to apply rotationally invariant technique as suggested in [10] and [7]. A new food source (child) from single abandonment solution $X^i|_G = X^i_1|_G, X^i_2|_G, X^i_3|_G, ..., X^i_D|_G$ is formed from binary crossover as equation 5 when Cr is up to 0.95.

  $$X^i_d|_G = \begin{cases} U^i, & \text{if } rand(0,1) > \text{Cr} \\ abd^i_d|_G, & \text{else if Cr} > 0.75 \\ v^i_d|_G, & \text{otherwise} \end{cases} \tag{5}$$

  Meanwhile, when the Cr is higher than or equal 0.95, the new food solution is considered as equation 6.

  $$X^i|_G = abd^i|_G + \psi(v^i|_G - abd^i|_G) \tag{6}$$

- **Selection.** The new food source generated from abandonment solution is supposed to be the new solution from the search space.

  Thus, selection is responsible to validate that new food source must consist at least one decision variable from the parent (abandonment solution) and also mutant. Further, the new solution must not be duplicated from the parent. After selection process, the new food sources will replace the current food sources for each ith abandonment agent and the cycle of improvement will be continued as depicted in Figure 1.

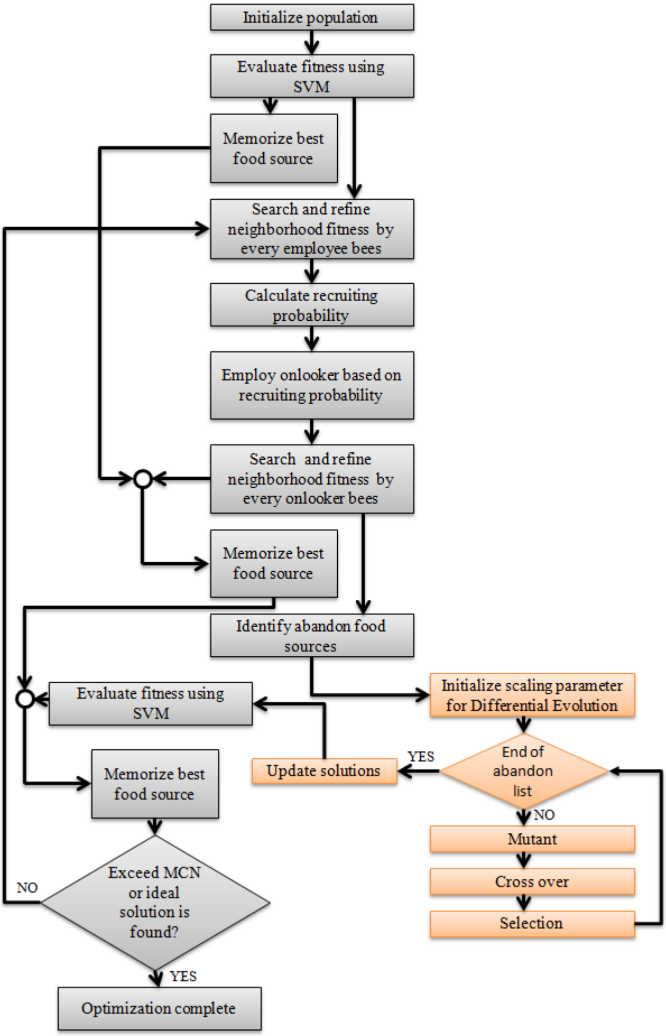

Initialize population
Evaluate fitness using SVM
Memorize best food source
Search and refine neighborhood fitness by every employee bees
Calculate recruiting probability
Employ onlooker based on recruiting probability
Search and refine neighborhood fitness by every onlooker bees
Memorize best food source
Identify abandon food sources
Initialize scaling parameter for Differential Evolution
End of abandon list
YES
NO
Mutant
Cross over
Selection
Update solutions
Evaluate fitness using SVM
Memorize best food source
Exceed MCN or ideal solution is found?
NO
YES
Optimization complete

Fig. 1. Flow chart of deABC algorithm for feature selection

3 Methodology

In the previous study [5], we had discussed the role of ABC with neighbourhood search modification as feature selection for ovarian cancer dataset. The result was promising in discriminating the cancer and healthy sample for biomarker discovery purposes. Thus in this study, we extended the experiment by hybridising the previous ABC with DE and compared the evaluation with another two popular bio-inspired algorithms. We provided the details on the implementation in every sub-section as followed.

3.1 Data Description

For the best implementation, two real world datasets of high-resolution mass spectrometry data were used. Both datasets were downloaded from National Cancer Institute (home.ccr.cancer.gov/ncifdaproteomics/ppatterns.asp).

Drug-Induced Toxicity (TOX): Analysis of serum from rat models of anthracycline and anthracenedione induced cardiotoxicity had been conducted by petricoin. In this study, only 34 control groups and 28 induced cardiac toxicities samples that exhibited definite range of positive and negative were selected to undergo analysis. Further, this dataset was be used to identify biomarkers that measured effects of therapeutic compounds on cardiac damage.

Ovarian Cancer Dataset: This high dimensional ovarian cancer dataset was developed based on QAQC procedure. These samples comprised 121 cancer samples, and 95 normal samples.

3.2 Pre-treatment

This study has followed [11] in applying pre-treatment methods for both TOX and ovarian cancer datasets. The treatment was carried out consecutively by starting from baseline removal, normalisation and noise filtering. Further, features were identified through peak detection method and extracted potential peaks that exhibit high discriminant power. Discriminative characters among peaks were evaluated through shrinkage covariance estimator since it has been proven by the previous study [5] for ovarian cancer dataset. In addition, we extended the shrinkage method for TOX dataset. Several peaks that exhibited strong correlation (more than 0.80) among features had been grouped as peaks-windows or peak-bin [2]. In order to select only around 400 most potential peaks-windows, this study combined statistical evaluation of strong discriminant analysis with quantified cumulative peaks across samples. All the peaks-windows had been sorted based on their total of peaks quantified across samples. Thus, the most popular peaks-windows were listed on top.

3.3 Feature Selection and Analysis

Selecting parsimonious features was the most crucial process in biomarker discovery. Prior feature selection, both datasets are split in ratio of 70:30 of training and testing respectively. The algorithm starts by initializing all the parameters related to deABC, and assigned all 400 extracted features as food sources among 50 agents of employee bees. Each agent was responsible to collect D food sources and optimize the nectar quality. Details of the implementation had been discussed in section 2. We used linear SVM classification (build-in MATLAB function) to evaluate the features in terms of nectar quality of food sources. More specifically, these tools had been used to evaluate the error rate of classification among features collected from particular agent of bee. SVM classification also was integrated with five times cross-validation procedure for optimal regularised cost function for each different agent.

Meanwhile for Ant Colony Optimisation (ACO) and Particle Swarm Optimisation (PSO), we followed the same programs and parameters setting that are available in [8] and [9] respectively. These parameters were based on equation below:

Ant Colony Optimisation(ACO)

Probability function, $P_i(t)$

$$P_i(t) = \frac{(\tau_i(t))^{\alpha}\eta_i^{\beta}}{\sum_i(\tau_i(t))^{\alpha}\eta_i^{\beta}} \tag{7}$$

Update pheromone trail,$\tau_i(t+1)$

$$\tau_i(t+1) = \rho.\tau_i(t) + \Delta\tau_i(t) \tag{8}$$

Where, α and β are both parameters that determine the relative influence of pheromone trail at time t, $\tau_i(t)$ and prior information, η_i . Meanwhile ρ is a parameter that represents evaporation of pheromone trail.

Particle Swarm Optimisation (PSO):

New trajectories, $\Delta\overrightarrow{x_i}(t+1)$

$$\Delta\overrightarrow{x_i}(t+1) = \chi(\Delta\overrightarrow{x_i}(t) + \phi_1(\overrightarrow{x_{i,best}}(t) - \overrightarrow{x_i}(t)) + \phi_2(\overrightarrow{x_{G,best}}(t) - \overrightarrow{x_i}(t))) \tag{9}$$

Update particle position, $\overrightarrow{x_i}(t+1)$

$$\overrightarrow{x_i}(t+1) = \overrightarrow{x_i}(t) + \Delta\overrightarrow{x_i}(t+1) \tag{10}$$

The trajectory vector,$\Delta\overrightarrow{x_i}(t)$ denotes the direction of motion in the search space at t^{th} iteration. Where, ϕ_1 and ϕ_2 are parameters that weigh the movement of particles in the direction of individual's best positions and global best positions, respectively. Meanwhile, constriction factor,χ is to ensure convergence of the PSO algorithm.

The extracted potential features from both TOX and ovarian datasets were fed and initialised to PSO and ACO algorithms by following the same procedure as deABC. We ran all of the algorithms 500 times (MCN) before the parsimonious features were analysed and finalised from the output of training and testing.

4 Result and Discussion

For the first dataset TOX, m/z windows in table 3 signified the most occurrence and parsimonious features that represent potential biomarkers for the drug-induced cardio toxicity. Several overlapping features were highlighted to show that particular features had been chosen repeatedly by different algorithms

Table 1. Parameters setting for deABC

Population size	50	**Max cross over rate**	1
Employee bees	50	**Median cross over rate**	0.5
Onlooker bees	100	**Min cross over rate**	0.2
Limit	100	**Max scaling factor**	0.5
MCN	500	**Median scaling factor**	0.4
recombination	0.75	**Min scaling factor**	0.3

Table 2. Parameters setting for ACO and PSO

ACO		PSO	
Number of ant	50	Number of particles	50
α	1	ϕ_1	0.5
β	1	ϕ_2	0.2
ρ	0.1	χ	0.5

Table 3. The eight most occurance m/z values from TOX datasets

	ACO		PSO		deABC	
	training	testing	training	testing	training	testing
Accuracy	97.73	66.67	100	83.33	100	94.44
Sensitivity	100	57.14	100	72.73	100	88.89
Specificity	0.96	100	100	100	100	100
m/z windows	4576.23-4601.332		3845.633-3863.987		3351.756-3398.349	
	5598.334-5603.099		4576.23-4601.332		**4295.566-4299.34**	
	5966.923-6010.033		**4295.566-4299.34**		3899.756-3905.944	
	7734.011-7736.608		6681.545-6695.073		**5598.334-5603.099**	
	8322.671-8390.456		8322.671-8390.456		7778.443-7788.983	
	9721.677-9799.011		7634.675-7677.689		**9931.899-9957.453**	
	10455.911-10479.877		**9931.899-9957.453**		10421.771-10435.956	
	10633.677-10671.532		10076.45-10099.65		**10633.677-10671.532**	

Table 4. The eight most occurance m/z values from Ovarian Cancer datasets

	ACO		PSO		deABC	
	training	testing	training	testing	training	testing
Accuracy	99.34	92.19	100	089.06	100	98.44
Sensitivity	98.53	100	100	100	100	96.55
Specificity	100	87.80	100	83.72	100	100
m/z windows	5129.705-5135.238		2001.152-2001.536		3860.689-3863.356	
	7189.546-7199.009		**7067.093-7070.701**		3893.821-3895.963	
	7067.093-7070.701		**7721.58-7736.608**		4299.631-4304.698	
	7721.58-7736.608		**7923.418-7925.71**		**7067.093-7070.701**	
	7923.418-7925.71		8896.567-8903.854		**7721.518-7736.608**	
	8597.99-8609.134		8931.409-8938.71		**7923.418-7925.71**	
	8704.954-8716.969		9356.479-9363.952		8692.148-8692.948	
	8931.409-8938.71		11709.78-11710.71		**8704.954-8716.969**	

and could be biologically significant as biomarkers. Anyhow, for TOX dataset no similar features were selected concurrently from those three different algorithms, where features of 4295.566-4299.34 and 9931.899-9957.453 were selected by deABC and PSO. Meanwhile features of 9931.899-9957.453 and 10633.677-10671.532 were selected by deABC and ACO algorithms as feature selection. Despite of that, the second dataset of ovarian cancer had exhibited three features from most occurrence m/z windows that overlapping for these three different algorithms. The three features are 7067.093-7070.701, 7721.518-7736.608 and 7923.418-7925.71. Further, features 8704.954-8716.969 was selected by deABC and ACO, meanwhile features 8931.409-8938.71 was selected by ACO and PSO. Compared to TOX, ovarian dataset had exhibited consistent and almost fair features occurrence among the three different types of feature selection. The deABC again produced better results in term of accuracy, sensitivity and specificity. The results of deABC had also exhibited better results and convergence compared to the ABC constructed in previous study [5].

5 Conclusion

ABC algorithm is easy to be implemented due to its simplicity and small numbers of control parameters. Anyhow simple enhancement is useful in order to improve its exploitation over the search space. Thus, simple enhancement using DE that popular with good exploitation had been applied. This study used two different types of dataset to evaluate the robustness and potential of deABC where simple ABC and DE were hybridized. Both analyses of table 3 and 4 have shown that deABC capable in producing superior results compared to ACO and PSO.

This study will be further extended to compare the deABC with others evolutionary and bio-inspired algorithms. After all, the assessment will also be extended to other statistical test, complexity calculation and discriminant analysis of the biomarkers between cancer and normal sample to evaluate its robustness. Apart from that, this simple combination could also be applied for other domain of problems in feature selection purposes.

Acknowledgement. We wish to express our heartfelt gratitude to the Ministry of Higher Education (MOHE) for funded grant under Fundamental Research Grant Scheme (FRGS) (203/PKOMP/6711268). Lastly, the first author would like to thank Universiti Teknologi MARA (UiTM) for offering a generous Ph.D. scholarship.

References

1. Al-Ani Ahmed. Feature subset selection using ant colony optimization (2005)
2. Armananzas, R., Saeys, Y., Inza, I., Garcia-Torres, M., Bielza, C., Van de Peer, Y., Larranaga, P.: Peakbin selection in mass spectrometry data using a consensus approach with estimation of distribution algorithms. IEEE/ACM Transactions on Computational Biology and Bioinformatics 8(3), 760–774 (2011)

3. Baykasoglu, A., Ozbakir, L., Tapkan, P.: Artificial bee colony algorithm and its application to generalized assignment problem. Swarm Intelligence: Focus on Ant and Particle Swarm Optimization, 113–144 (2007)
4. Kennedy, J.: Particle swarm optimization. In: Encyclopedia of Machine Learning, pp. 760–766. Springer (2010)
5. Mohamed Yusoff, S.A., Abdullah, R., Venkat, I.: Using ABC algorithm with shrinkage estimator to identify biomarkers of ovarian cancer from mass spectrometry analysis. In: Pan, J.-S., Polycarpou, M.M., Woźniak, M., de Carvalho, A.C.P.L.F., Quintián, H., Corchado, E. (eds.) HAIS 2013. LNCS, vol. 8073, pp. 345–355. Springer, Heidelberg (2013)
6. Mohd Shukran, M.A., Chung, Y.Y., Yeh, W.C., Wahid, N., Ahmad Zaidi, A.M.: Artificial bee colony based data mining algorithms for classification tasks. Modern Applied Science 5(4), 217 (2011)
7. Price, K.V.: An introduction to differential evolution. In: New Ideas in Optimization, pp. 79–108. McGraw-Hill Ltd., UK (1999)
8. Ressom, H.W., Varghese, R.S., Drake, S.K., Hortin, G.L., Abdel-Hamid, M., Loffredo, C.A., Goldman, R.: Peak selection from maldi-tof mass spectra using ant colony optimization. Bioinformatics 23(5), 619–626 (2007)
9. Ressom, H.W., Varghese, R.S., Abdel-Hamid, M., Eissa, S.A.-L., Saha, D., Goldman, L., Petricoin, E.F., Conrads, T.P., Veenstra, T.D., Loffredo, C.A., Goldman, R.: Analysis of mass spectral serum profiles for biomarker selection. Bioinformatics 21(21), 4039–4045 (2005)
10. Reynoso-Meza, G., Sanchis, J., Blasco, X., Herrero, J.M.: Hybrid de algorithm with adaptive crossover operator for solving real-world numerical optimization problems. In: 2011 IEEE Congress on Evolutionary Computation (CEC), pp. 1551–1556 (2011)
11. SyarifahAdilah, M., Abdullah, R., Venkat, I.: Abc algorithm as feature selection for biomarker discovery in mass spectrometry analysis. In: 2012 4th Conference on Data Mining and Optimization (DMO), pp. 67–72. IEEE (2012)
12. Tu, C.-J., Chuang, L.-Y., Chang, J.-Y., Yang, C.-H., et al.: Feature selection using pso-svm. IAENG International Journal of Computer Science 33(1), 111–116 (2007)
13. Yusoff, S.A.M., Venkat, I., Yusof, U.K., Abdullah, R.: Bio-inspired metaheuristic optimization algorithms for biomarker identification in mass spectrometry analysis. International Journal of Natural Computing Research (IJNCR) 3(2), 64–85 (2012)

An Artificial Intelligence Technique for Prevent Black Hole Attacks in MANET

Khalil I. Ghathwan[1,2] and Abdul Razak B. Yaakub[1]

[1] InterNetWorks Research Group, School of Computing, College of Arts and Sciences,
Universiti Utara Malaysia, 06010 Sintok, Kedah, Malaysia
s93453@student.uum.edu.my, ary321@uum.edu.my
[2] Computer Science Dept., University of Technology, Baghdad, Iraq
k.i.ghathwan@gmail.com

Abstract. Mobile ad hoc networks (MANETs) can be operated in the difficult environments or emergency situations. In this type of networks, the nodes work of forwarding packets together. Routing protocols are worked based on multi-hop to discover a path from source to destination node when the direct path between them does not exist. One of the standard MANET protocols is Ad hoc on-demand distance vector protocol (AODV). AODV is attacked by many types of attacks such as black hole attack due its routing mechanism. Black hole provides highest destination sequence number and lowest hop count number to attract sourcenode and drop the packets. Most previous works were used trusted neighbor nodes for preventing black hole attack and making AODV more secure. However, these solutions suffer from high routing overhead and missing specific mechanism for providing a shortest secure path. In this paper, we propose an intelligent preventing technique for AODV to prevent black hole attacks, which is called Shortest Secure Path for AODV (SSP-AODV). This intelligent technique is integrated A* and Floyd-Warshall's algorithms. The simulation is conducted in Network Simulator 2. The results indicate that the proposed intelligent technique outperform standard AODV in two terms; packet loss delivery and average End-to-End delay. The performance of proposed technique can significantly reduce the effect of black hole attacks.

Keywords: Mobile Ad hoc Networks (MANETs), Black Hole attack, Heuristic Search, A* algorithm, Floyd-Warshall's, Malicious node, Secure Routing.

1 Introduction

MANET wireless network is composed of several of mobile nodes. It can be worked in a practical environment without any need of backbone infrastructure[1]. MANETs have many applications in different area such as emergency operations, Military, civilian environments and personal area networking [2]. On the other hand, it suffers from many limitations such as shortbattery lifetime, limited capacities and malicious behaviors. A black hole is a severe attack that is exploited in MANETs. It attacks the routing protocol in the network to drop the network. Most the famous MANET protocols like AODV [3], the main goal of routing finds the path to the destination

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_12, © Springer International Publishing Switzerland 2014

node. It uses a route request (RREQ), to routing discovery and a route reply (RREP), to replay. According to the routing mechanism in AODV, a black hole node can be adjusted to the values of (hop count) and (DSN) easily in order to deceive the source node. In this attack, the source node after sending a route request (RREQ), it will respond to the first route reply (RREP), that will be coming from the black hole node and not reply to other intermediate nodes. As a result, it will be terminated the cooperation work in MANET [1],[6],[7],[8] and [9].

However, to make AODV routing protocol more secure, we need to find not a shortest only but a shortest secure path. furthermore, the solution with a high overhead are worthless. Also, a complex algorithm that needs high calculation among neighboring nodes may to discharging the limited battery power.

In other words, we need to find a solution that can find a shortest secure path without need more high overhead. In this paper, we proposed a new technique (SSP-AODV) uses a heuristic search algorithm A* [4][5] and Floyd-Warshall's algorithm [5] to avoid black hole attack on AODV. Particularly, SSP-AODV and (AODV) are same protocol but, with some changes in the original AODV to avoid black hole nodes. Floyd-Warshall's algorithm will work synchronously with RREQ and RREP. It is used to calculate the distances between the neighboring nodes. Nevertheless, the Floyd-Warshall's algorithm cannot be used to find the shortest path from source to destination nodes. Therefore, we will use A* a heuristic search algorithm to find the shortest path between the source and destination nodes. With this technique, A* algorithm and Floyd-Warshall’s algorithm will work together as a new simple mechanism to reduce the complex calculation security overhead and encase a security to routing algorithm.

The next sections of this paper are arranged as follows, Section 2 discusses some related works, Section 5 present the proposed solution and Section 6 discusses the experiment setup, results and analysis. Section 7 concludes the paper.

2 Related Works

Some approaches to detect and defend against a black hole attack in MANET was proposed [6], [7], [8] and [9].

In [6] a solution was proposed for cooperative black hole attack in MANET. It makes some modifying for original AOD, when a source node receives the first RREP, it does not respond directly (respond to send the packet), but it waits for a specific time. A source node has a cache memory to save all RREP and all details of next hop that gather from other nodes. It chooses the correct path from a list of response paths after checking for repeating next hop node, otherwise it chooses a random path. The important point of this solution is the suggestion that solves the collective black hole and makes the fidelity tables. However, it is not an accurate way to catch a black hole attack and reduce the delay in the whole network because the exchanges of fidelity packets increase the control overhead.

In [7] a new black hole detection method was proposed. It is based on the fact that the attackers relied on changing the destination sequence number to the maximum number. Consequently, the attackers will gain the routing and drop the packets. The authors made a statistical anomaly detection rule to detect the black hole depending on the difference among DSN in receiving a RREP. This technique can detect black hole in AODV with low overhead but the false positive is a main drawback of this proposed.

In their work [8] the authors suggested a new method to detect black hole attacks in AODV protocol during the routing discovery. In this method, when the intermediate node receives a RREQ from source node they send a new RREP to destination node which is called SREP. SREP has special field to save the current sequence source number (SSN), which it gathers from the destination node. The exchanges of RREQ, RREP and SRRP between source and destination nodes in the whole network leads to increase the control overhead. Probably, this method is noteffective in large topology that has a high mobility.

In [9] a new algorithm for intrusion detection was proposed. It is a distributed collaborative approach in ad hoc wireless networks. In this algorithm, each node is locally and independently IDS work, but the nearby nodes work together to monitor a larger area. Each node is responsible for overseeing the activities of the local data, If ananomaly is detected in the local data, or if the evidence is not sufficient and requires a more comprehensive search, neighboring IDS agents cooperate to realize the global intrusion detection. This work is focused on a new proposed algorithm that make a trusted neighbor node, in order to add a security in AODV. However, this algorithm has a high routing control overhead.

3 Floyd-Warshall's Algorithm

Floyd-Warshall's algorithm can be used to determine the length of the shortest path between two nodes in any graphnet. In the network by using the values of a weighted connection between them. In spite of this, Floyd-Warshall's algorithm cannot be used to find the shortest path such as heuristic search algorithms. Figure 1 shows the Floyd-Warshall's algorithm pseudocode [5].

```
procedure Floyd-Warshall (intn,int w[1..Node i,1..Node
j])
array d[1..n, 1..n]
fori = 1 to n do                  // Phase One
for j = 1 to n do
d[Node i, Node j] = w[Node i, Node j]
d[Node i, Node j] = •
fori = 1 to n  do            // Phase Two
for j = 1 to n  do
for k = 1 to n do
if(d[Node j,Node i]+d[Node i,Node k]<d[Node j, Node k])
                then{
d[Node j, Node k]= d[Node j, Node i]+
d[Node i, Node k]
     d[Node i, Node j] = i
     end.
```

Fig. 1. Floyd-Warshall's algorithm pseudocode

4 Heuristic Search Algorithm A*1

A* heuristic search algorithm is a graph search algorithm that finds the shortest pathfrom source to destination nodes. It has objective functionf(n) that calculate the estimated total cost of a path through n to the goal. This objective function f(n) needs two factors for calculating which are; g(n) and h(n). Where g(n) is the cost so far to reach n, h(n) is the estimated cost from n to the goal. equation 1 shows the objective function of A* algorithm.

$$f(n) = g(n) + h(n) \tag{1}$$

5 The Proposed SSP-AODV

In order to address the problem that was described in section 2, we propose an intelligent preventing technique for AODV to prevent black hole attacks(SSP-AODV). In AODV, each node has a routing table which includes the information such as; hop count, destination sequence number (DSN), life time, source IP address and so on. Every node can be calculated the estimate time by using this information in routing table. We add two fields in routing table which are Estimate time and Shortest path (SP-value) as shown in Figure 2.

RREQ-AODV Table	RREP-AODV Table	SSP-AODV Table
Broadcast ID		Destination IP & DSN
Destination IP Address	Destination IP Address	DSN-Flags
Destination Sequence Number	Destination Sequence Number	Flags
Source IP Address	Source IP Address	Network Interface
Source Sequence Number	Life Time	Hop Count
Hop Count	Hop Count	Next Hop
Estimated Time	Estimated Time	Life Time
SP- Value	SP- Value	Best Path to Destination

Fig. 2. Routing table of SSP-AODV protocol in RREQ,RREP and Routing Table

The Estimate time equation is calculated by using equation 2 as shown below.

$$ET(n) = \frac{SN(n)}{\sqrt{d(n)} * d(n)} \tag{2}$$

Where, ET(n) = Estimated Time from node n to destination node, SN (n) = The Sequence Number of node n,d (n) The number of hop count of node. In this paper, we propose to integrate Floyd-Warshall's Algorithm with A* search algorithm to find shortest path respectively, we check for secure path. Figure 3, shows the flowchart of the propose SSP-AODV.

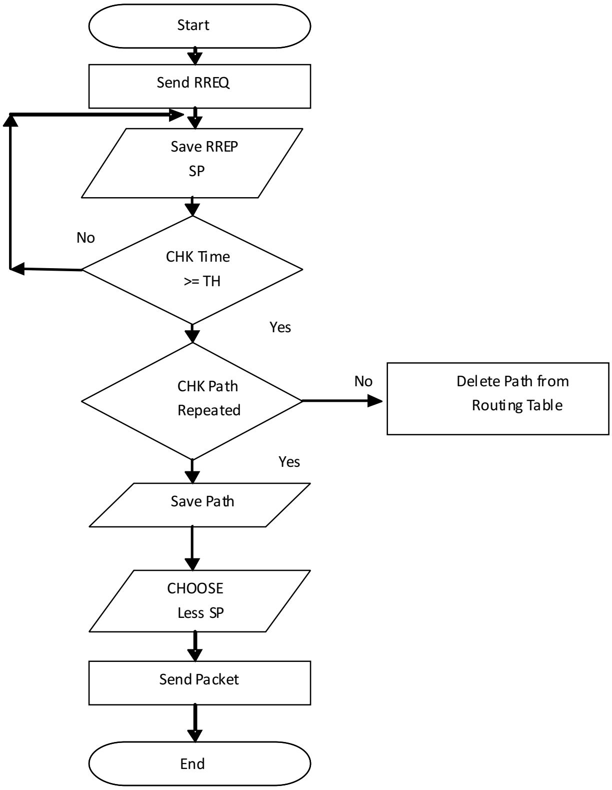


Fig. 3. Flowchart of SSP-AODV preventing technique

The preventing technique SSP-AODV has two phases; Calculation the shortest path and checking for secure path.

5.1 PhaseOne: Shortest Path

In this phase, we use Floyed-Warshall's algorithm to find the value of the shortest path. In Figure 4, we assume node (a) is a secure node and node (f) is a destination node, the b,c, and e are intermediate nodes.

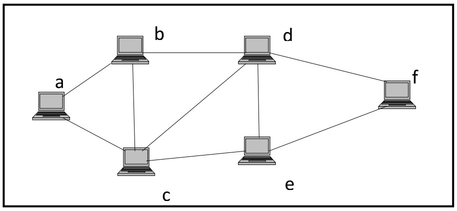


Fig. 4. Topology example of six nodes (node (a) is a source node, node (f) is a destination node and nodes b, c, d & e are intermediate nodes)

The value of estimated time for each node to destination node is calculated using equation 1. The table 1 shows the value of estimated time for each node of route discovery to destination node (f).

Table 1. Example of Estimated Time ofroute discovery

Source Nodes	Estimated Time(sec.)
a	5.0
b	3.0
c	3.5
d	1.5
e	2.0
f	0.0

The distance between two nodes is calculated using the equation 3 as shown below.

$$ED(n1, n2) = ET(n1) + ET(n2) \tag{3}$$

Where, ED is estimateddistance between two nodes, ET is the estimated time of the node. Figure 5 shows the topologyexample of six nodes with the estimated distance between node.

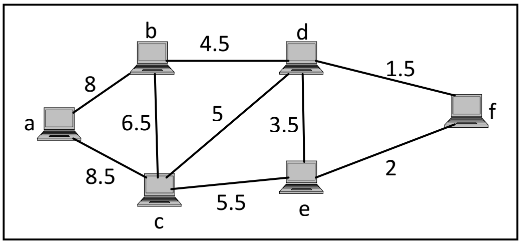


Fig. 5. Topology example of six nodes with the estimated distance

$$\begin{pmatrix} \infty & 8 & 8.5 & \infty & \infty & \infty \\ 8 & \infty & 6.5 & 4.5 & \infty & \infty \\ 8.5 & 6.5 & \infty & 5 & 5.5 & \infty \\ \infty & 4.5 & 5 & \infty & 3.5 & 1.5 \\ \infty & \infty & 5.5 & 3.5 & \infty & 2 \\ \infty & \infty & \infty & 1.5 & 2 & \infty \end{pmatrix} \Longrightarrow \begin{pmatrix} 16 & 8 & 8.5 & 12.5 & 16 & 14 \\ 8 & 8.5 & 6.5 & 4.5 & 12 & 6 \\ 8.5 & 6.5 & 10 & 5 & 5.5 & 6.5 \\ 12.5 & 4.5 & 5 & 3 & 3.5 & 1.5 \\ 16 & 12 & 5.5 & 3.5 & 4 & 2 \\ 14 & 6 & 6.5 & 1.5 & 2 & 3 \end{pmatrix}$$

(a-First Phase) (a-Final Phase)

Fig. 6. The two phases of applying Floyd-Warshall's algorithm

After that, as in Figure 1 we did the implementation of Floyd-Warshall'salgorithm to calculate the final distance between all nodes. Figure 6 shows the first phase and final phase of applying Floyd-Warshall's algorithm. the first phase of the algorithm which is a matrix (6x6); after six iterations we find the value of shortest pathas shown in the final phase.

Now, we implement the A* using equation 1. We suppose g(n) is the estimate distance between two nodes, h(n) is estimated cost to the goal which can derive from Floyd-Warshall's algorithm.

f(n) = {*the cost to reach n*} **g(n)** + {*estimate cost to the goal*} **h(n)** {*find from Floyd-Warshall's algorithm*}
a -> b
f(b)=g(b)+h(b) ; => 8.0+6.0 => 14.0
a -> c
f(c)=g(c)+h(c) ; => 8.5+6.0 => 15.0
So we choose node b.

5.2 Phase Two: Prevent Black Hole Attack

The phase two is the working on preventing the black hole attack, as shows in Figure 3 when the source node receives one or more RREP from the neighboring nodes, it will not send a packet directly but it waits until the routing time expires. The source node will wait for a short time, then, it will checkbased on threshold (TH = 10,20,…,100 second), if the waiting time is greater or equal than TH, then will check the hop count value of the new path with all RREP/RREQtables. The new path save as a secure path if it has the same hop count with tables, else the path remove from the routing table. The shortest secure path will choose from SP-value fields of RREQ/RREQ tables. Finally, the packet will send to the destination node.

6 Experiments Setup, Results and Analysis

We use NS2 simulator version 2.33[10] to experiment of three scenarios. Scenario 1 is to test the original AODV, scenario 2 is to test the black hole AODV and scenario 3 is to test the execution of the proposed SSP-AODV for finding the shortest secure path and securing the AODV protocol. The Simulation Parameters for scenario 1,2 and 3 are shown in Table 2.

6.1 Performance Metrics

Three performance indicators measure the performance of the proposed algorithm, original AODV, black hole AODV and SSP-AODV. These performance metrics explain in the following steps:

Table 2. Simulation Parameters for Scenarios 1,2,3

Parameter	Simulation1	Simulation2	Simulation3
Simulation Time	1000 sec.	1000 sec.	1000 sec.
Number of Nodes	50	50	50
Routing Protocol	AODV	BlackHole-AODV	SSP-AODV
Traffic Model	CBR(UDP)	CBR(UDP)	CBR(UDP)
Pause Time	2 sec.	2 sec.	2 sec.
Maximum Mobility	60 m/sec.	60 m/sec.	60 m/sec.
No. of sources	1	1	1
Map area	800m x 800m	800m x 800m	800m x 800m
Transmission Range	250m	250m	250m
Malicious Node	1	1	1

- End-to-End delay (φ): The average time taken for a data packet to reach the destination. And also includes the delay caused by the process of route discovery response and the tail makes transmission of data packets. Only the data packets successfully addressed and delivered are counted. Equation 4 is shown the End-to-End delay.

$$\varphi = \frac{\sum(\alpha - \beta)}{\sum(\delta)} \tag{4}$$

Where, α is arrival time, β is transmission time, δ is the number of connections. The lower the end to end delay value is better the performance of the protocol.

- Packet loss (τ): The total number of packets lost during the simulation is computed using equation 5.

$$\tau = [\sum(\mu) - \sum(\vartheta)] * (\frac{100}{\sum(\mu)}) \tag{5}$$

Where μ is number of packets send, ϑ is number of packets received. The lower value of the package loss is better the performance of the protocol.

- Packet delivery ratio (*PDR*) defines the ratio of the number of data packets delivered to the destination. This metric shows the amount of data that arrived at the destination. Equation 6 is shows the packet delivery ratio.

$$PDR = \frac{\sum(\vartheta)}{\sum(\epsilon)} \tag{6}$$

Where ϵ is the number of packets. The largest package delivery ratio means better performance of the protocol.

6.2 Packet Loss: Results and Discussion

In Figure 7 three scenarios; original AODV, blackhole AODV and SSP-AODV are compared. The packet loss ratio increases in black hole AODV which degrades the performance of the protocol and causes many packet losses. As a result, it triggers a DoS attack in MANET. The proposed SSP-AODV is minimizing the packet loss and improves the network performance compare than original AODV. Packet loss is 21.41% in AODV, but it increases in black hole AODV which is 28.32%, after implementing SSP-AODV the percentage is improved which is 22.98%.

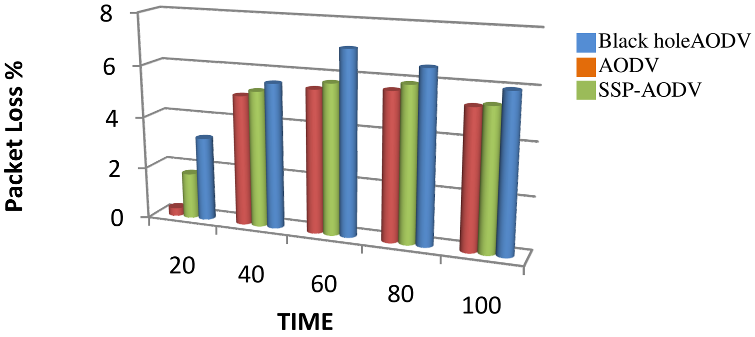


Fig. 7. Packet Loss Percentage for AODV, Black Hole AODV and SSP-AODV

6.3 Average End-to-End Delay: Results and discussion

In Figure 8, the comparison of the average End-to-End delay of the three scenarios is shown. The average End-to-End Delay increases with the existing of a black hole. This delay degrades the performance of the network and causes more delay time when packets try to reach the destination node. Furthermore, when we compare the original AODV with the proposed protocol SSP-AODV, the result indicated that SSP-AODV minimizes the Average End-to-End Delay and improves the network performance. The percentage of delay is 29% with black hole AODV comparing with original AODV. This percentage is about 11.42% with SSP-AODV.

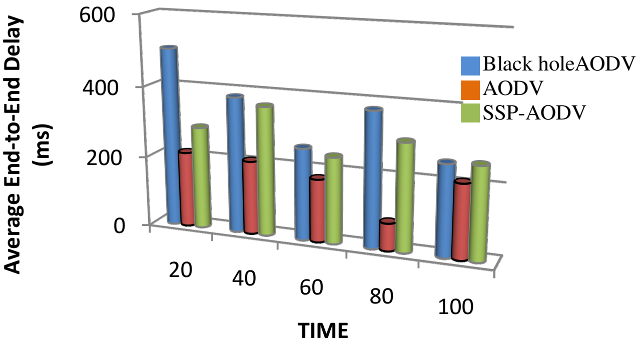


Fig. 8. The average end-to-end delay for AODV, Black Hole AODV and SSP-AODV

6.4 Packet Delivery Ratio (PDR): Results and Discussion

In Figure 9,the Packet Delivery Ratio for three scenarios are shown. We can see from the figure that the packet delivery ratio does not increase with the existing of the black hole in the network. The packets were reached to destination from source node was 479.77 in total for standard AODV, 469.56 for AODV with black hole nodes and 470.82forSSP-AODV. So we can see that the overall PDR of SSP-AODV does not degrade significantly due to the implementation of security algorithm.

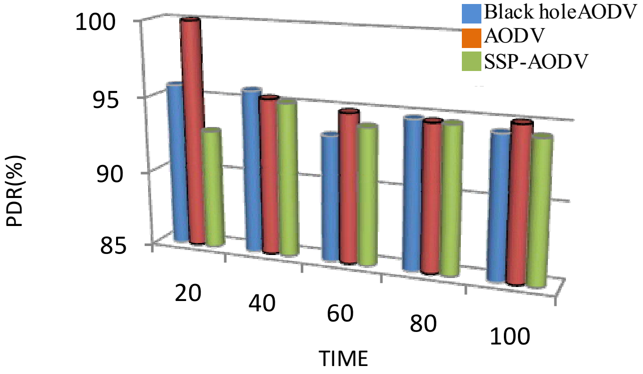


Fig. 9. The Packet Delivery Ratio for AODV, Black Hole AODV and SSP-AODV

7 Conclusion

This paper proposes a defence mechanism against a cooperative black hole attack in a MANET that relies on AODV routing protocol named as SSP-AODV Protocol. The proposed SSP-AODV modifies the standard AODV and optimizes the routing process by incorporating two techniques A* search algorithm and Floyd-Warshall's algorithm in the AODV routing process. The technique A* algorithm and Floyd-Warshall's algorithm uses the value of hop count and the estimate time as input. New routing mechanism and some changes in routing tables are used to delay RREP and checking repeated paths. The experimental results show that SSP-AODV is able to improve the performance of the network in the two metrics while securing from black hole attack.For future work we plan to consider implementation of more complex black hole attacks, as well as, other routing protocols such as DSR, CBRP, ZRP.

References

1. Tseng, F.-H., Li-Der, C., Han-Chieh, C.: A survey of black hole attacks in wireless mobile ad hoc networks. Human-Centric Computing and Information Sciences 1(1), 1–16 (2011)
2. Burbank, J.L., Chimento, P.F., Haberman, B.K., Kasch, W.: Key Challenges of Military Tactical Networking and the Elusive Promise of MANET Technology. IEEE Communications Magazine 44(11), 39–45 (2006), doi:10.1109/COM-M.2006.248156

3. Perkins, C., Royer, E.: Ad hoc on demand distance vector Routing. In: Proceeding of the second IEEE Workshop on Mobile Computing Systems and Applications, New Orleans (1999)
4. Edelkamp, S., Schrodl, S.: Heuristic Search: Theory and applications. Morgan Kaufmann Publishers, Elsevier, USA (2012)
5. Chen, W.K.: Theory of Nets: Flows in Networks. A Wiley-Interscience Publication, USA (1990)
6. Tamilselvan, L., Sankaranarayanan, V.: Prevention of Co-operative Black Hole Attack in MANET. In: The 2nd International Conference on Wireless Broadband and Ultra Wideband Communications, AusWireless, pp. 21–26 (2008)
7. Kurosawa, S., Nakayama, H., Kato, N., Jamalipour, N., Nemoto, Y.: Detecting Blackhole Attack on AODV-Based Mobile Ad Hoc Networks by Dynamic Learning Method. International Journal of Network Security 5(3), 338–346 (2007)
8. Zhang, X.Y., Sekiya, Y., Wakahara, Y.: Proposal of a method todetect black hole attack in MANET. In: International Symposium on Autonomous Decentralized Systems, ISADS 2009, vol. 1(6), pp. 23–25 (2009), doi:10.1109/ISADS.2009.5207339
9. Zhang, X.Y., Lee, W.: Intrusion Detection In Wireless Ad-Hoc Networks. In: Proceeding of Mobicom Conference, pp. 275–283 (2000), doi:10. 1145/345910.345958
10. Ns2 network simulation tools, `http://www.isi.edu/nsnam/ns/`

ANFIS Based Model for Bispectral Index Prediction

Jing Jing Chang[1], S. Syafiie[1], Raja Kamil Raja Ahmad[2], and Thiam Aun Lim[3]

[1] Department of Chemical and Environmental Engineering
syafiie@upm.edu.my
[2] Department of Electrical and Electronic Engineering
[3] Anaesthesiology Unit, Department of Surgery
Universiti Putra Malaysia, 43300 Serdang, Malaysia

Abstract. Prediction of depth of hypnosis is important in administering optimal anaesthesia during surgical procedure. However, the effect of anaesthetic drugs on human body is a nonlinear time variant system with large inter-patient variability. Such behaviours often caused limitation to the performance of conventional model. This paper explores the possibility of using the Adaptive Neuro-Fuzzy Inference System (ANFIS) to create a model for predicting Bispectral Index (BIS). BIS is a well-studied indicator of hypnotic level. Propofol infusion rate and past values of BIS were used as the input variables for modelling. Result shows that the ANFIS model is capable of predicting BIS very well.

Keywords: anesthesia, modeling, PKPD, neuro-fuzzy.

1 Introduction

Anaesthesia is a “clinical art” that involves calibration of observations of stimuli and responses against the dosage of anaesthetic drugs [1]. These observations include visible responses such as verbal responses, movement and respiration rate as well as advanced monitoring indicators derived from brain’s electrical activity or heart rate. Anaesthetic drugs are constantly adjusted according to these signals to achieve the desired anaesthetic depth. More specifically, anaesthetic agents are regulated to achieve adequate hypnosis and analgesia level.

Hypnosis and *analgesia (antinociception)* are two essential components of general anaesthesia. Hypnosis is associated with loss of consciousness and amnesia while analgesia leads to absence of pain. Propofol is a common hypnotic agent, often administered together with opioid, to induce desirable depth of anaesthesia.

The advancement of sensor and signal processing has resulted in rapid development and commercialization of many monitors that quantify depth of anaesthesia. These includes Bispectral Index (BIS) monitor, AEP-Monitor/2, NeuroSENSE, NarcoTrend, CSM, [2] etc. Among them, BIS is the most common electronic index for measurement of depth of hypnosis [3].

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_13, © Springer International Publishing Switzerland 2014

BIS is an empirical derived parameter resulted from the weighted sum of a composite of multiple sub-parameters (bispectral analysis, burst suppression and b-activation). From these sub-parameters, a single dimensionless index is created using multivariate statistical modelling. BIS ranges from 0 (isoelectric state) to 100 (fully awake). The recommended value of BIS during surgical is 40 to 60 [4].

The effect of propofol on BIS is usually described by pharmacokinetic and pharmacodynamic (PKPD) model, where PK explains the dose-concentration relationship while PD represents the nonlinear concentration-effect (BIS) relationship. However, the model parameters are developed based on population analysis and are subjected to large inter-patient variability.

Recently, the development of "individualized model" has gain much attention from various research group [5–8]. Parameters such as time delay and drug sensitivity are identified from past data, allowing the patient to have model parameters of his own. This approach potentially resolves variability among patients. However, parameters estimation will depends on the limited data collected in real-time during the initial phase of surgery. Furthermore, due to time-varying behaviour of the system, it is expected to require parameters tuning over time.

An alternative approach to model a complex system is through the application of soft computing. This technique does not require precise mathematical model. One such approach is the adaptive neuro-fuzzy inference system (ANFIS).

Conventional neural network modelling utilizes real plant data to detect trends that are too complex for other computer techniques. However, the "black box" model yielded is difficult to interpret. On the other hand, fuzzy logic system is capable of approximating any incomplete, imprecise or unreliable information to a set of constraints. ANFIS integrates the advantages of neural network and fuzzy-logic system, enabling the use of system input-output data to refine the fuzzy if-then rules through back propagation and least-squared training. ANFIS model has been proven to yield remarkable results in modelling multivariable nonlinear functions, identifying nonlinear components of an on-line control system as well as predicting chaotic time series [9, 10].

ANFIS is widely used in the area of medicine and healthcare due to its ability to handle complex biological system. In anaesthesia, it has been used for monitoring and measuring anaesthesia's depth [11–13], modelling [14, 15] and decision support system [16, 17].

This paper aims to develop an ANFIS model to predict BIS from propofol infusion rate and past BIS value. To the authors' best knowledge, this is the first paper on development of BIS prediction model from propofol infusion rate directly using ANFIS. An accurate propofol-BIS prediction model is useful for forecasting patient's BIS level during surgical procedure. It can also used for the design and evaluation of various control strategies.

The arrangement of this paper is as follows. Section 2 describes the ANFIS modelling. Results and discussion are presented on Section 3. Section 4 gives a concluding remarks.

2 ANFIS Modelling

In this study, three inputs were chosen: propofol infusion rate ($PPF(t-5)$), and past values of BIS ($BIS(t-5)$ and $BIS(t-10)$), where t is time in second. These inputs were selected based on *a priori* knowledge and sequential search. Figure 1 shows the structure of the model.

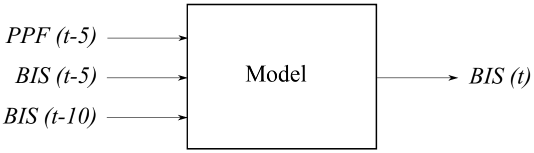


Fig. 1. Structure model for prediction of BIS

The two main tasks to model a fuzzy inference system are structure determination and parameter identification. Structure determination defines the relevant inputs, number of membership functions (MF) for each inputs, number of rules and types of fuzzy models. Parameter identification determines the values of parameters that can generate satisfactory performance.

2.1 Model Structure

Here, the input space was partitioned by grid partition. Each input was assigned with two bell-shaped MF. The number of MF and types of input MF were chosen by trial-and-error method. The output MF was of type linear.

The corresponding ANFIS architecture is as shown in Figure 1 where v, x and y represent $PPF(t-5)$, $BIS(t-5)$ and $BIS(t-10)$ respectively.

ANFIS architecture [9] consists of five layers where each layer contains a set of nodes connected through directed links as in Figure 1. Each node is a process unit that performs a particular node function on its input signal.

There are two types of node: adaptive node (represented by a square) and fixed node (denoted by a circle). Adaptive node contains parameterized function with modifiable parameters; fixed node contains a fixed function with empty parameter set. In the rest of the paper, the term $O_{l,i}$ denotes output node i^{th} of layer l.

1. *Layer 1*: In the first layer, input fuzzification takes place. Each input is assigned a membership value to each fuzzy set by the node function. The node output is expressed as

$$\begin{aligned} O_{1,i} &= \mu_{A_i}(v), && \text{for } i = 1,\ 2, \text{ or} \\ O_{1,i} &= \mu_{B_{i-2}}(x), && \text{for } i = 3,\ 4, \text{ or} \\ O_{1,i} &= \mu_{C_{i-4}}(y), && \text{for } i = 5,\ 6 \end{aligned} \tag{1}$$

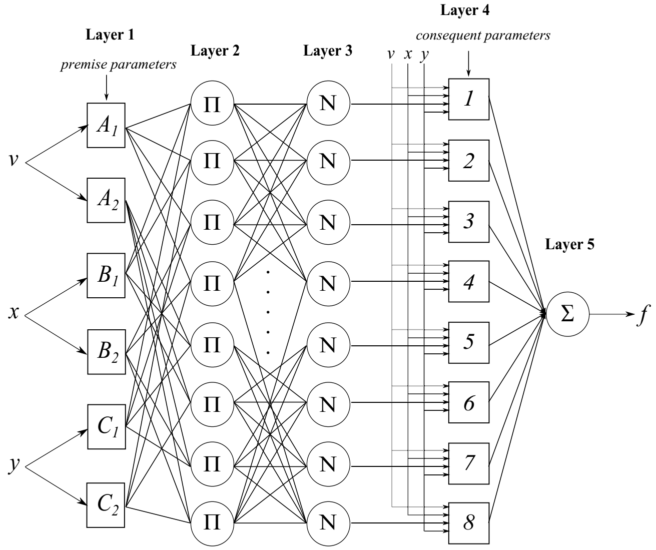


Fig. 2. "Grid"type ANFIS architecture for three inputs and one output

where μ_{A_i}, $\mu_{B_{i-2}}$ and $\mu_{C_{i-4}}$ represent the appropriate MF. Common MF includes bell-shaped MF, Gaussian MF, sigmoidal MF and triangular MF. For bell-shaped MF, $\mu_{A_i}(v)$ is characterised by

$$\mu_{A_i}(v) = \frac{1}{1 + [(\frac{v-c_i}{a_i})^2]^{b_i}} \tag{2}$$

The linguistic label A_i for input v will have a parameter set of $\{a_i, b_i, c_i\}$. These parameters are adjustable during the training of data. Parameters in this layer are also called premise parameters.

2. *Layer 2*: The nodes in this layer are fixed node (label *Π*) which multiplies the incoming signals. The output of the nodes, i.e. the product of multiplication, represents the firing strength of a rule.

$$O_{2,i} = w_i = \mu_{A_i}(v) \times \mu_{B_i}(x) \times \mu_{C_i}(y), \quad i = 1,\, 2. \tag{3}$$

3. *Layer 3*: The nodes in this layer are fixed node (label N) which calculates the ratio of the i^{th} rule's firing strength to the sum of all rules' firing strength. The output of this layer is also called normalised firing strength.

$$O_{3,i} = \tilde{w}_i = \frac{w_i}{w_1 + w_2}, \quad i = 1,\, 2. \tag{4}$$

4. *Layer 4*: The node output in this layer is expressed as

$$O_{4,i} = \tilde{w}_i f_i = \tilde{w}_i(p_i v + q_i x + r_i y + s_i) \tag{5}$$

where $\tilde{w}_i$ is the output of layer 3. The nodes in this layer are adaptive node, i.e. the parameter set $\{p_i, q_i, r_i, s_i\}$ are adjustable. Parameters in this layer are called consequent parameters.

5. *Layer 5*: This layer contains a single fixed node labeled $\sum$. The output of the node is the summation of all the incoming signals:

$$O_{5,i} = \sum_i \tilde{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{6}$$

The output of this node is the overall predicted output of the ANFIS model.

2.2 Model Training

Data collected from 3 patients in the intensive care unit (ICU) were used to train the ANFIS model. Relevant data are recorded every 5 seconds. Random disturbances were introduced to the patient during the data collection process. The patients studied were 65 ± 11 years with weight 95 ± 17 kg and height 173 ± 12 cm. All patients are male.

The actual propofol infusion rate and BIS value as recorded were shown in figure 3. The data for patient 1, 2 and 3 were shown in the first, second and third column of figure 3 respectively. The BIS data was preprocessed with low-pass filter before training.

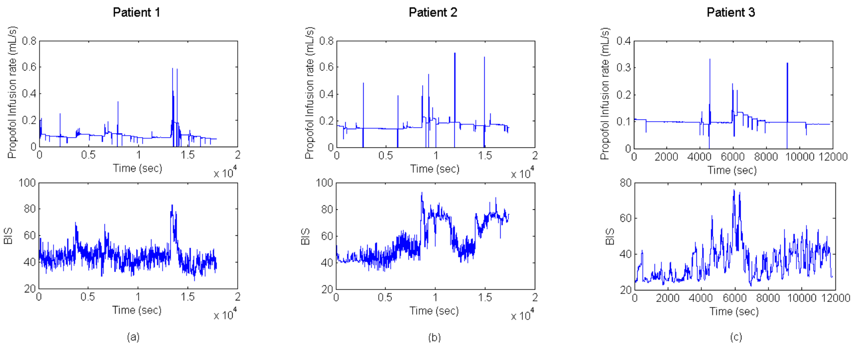


Fig. 3. Propofol infusion rate and BIS value recorded for (a)Patient 1, (b) Patient 2, and (c) Patient 3

Parameter identification was tackled by hybrid learning algorithm that combines the back-propagation gradient descent and the least squares method. The ANFIS model was built using `anfis` command in MATLAB.

The resulting ANFIS model for patient 2 is tabulated in Table 1 and 2. Input MF parameters as mentioned in (2) are listed in Table 1. Output MF parameters as mentioned in (5) are tabulated in Table 2. The model has 50 parameters and 8 rules. To avoid overfitting 14 epochs was used.

Figure 4 and 5 show the surface views of the input and output relationship.

Table 1. Input membership function parameters

Premise	v		x		y	
parameters	MF 1	MF 2	MF 1	MF 2	MF 1	MF 2
a	0.298	0.390	25.364	25.357	25.357	25.361
b	1.995	2.003	2.064	2.003	1.947	1.992
c	-0.078	0.661	39.361	90.075	39.347	90.072

Table 2. Output membership function parameters

Consequent	Rule number							
parameters	1	2	3	4	5	6	7	8
p	0.344	-3.852	0.196	1.111	-0.789	-11.887	10.836	-0.096
q	-0.893	-5.004	0.187	-0.958	-0.905	4.769	-5.799	-0.731
r	1.887	5.400	1.339	1.952	1.878	-2.953	5.666	1.716
s	0.032	21.346	-15.748	0.163	0.865	-37.935	59.800	2.044

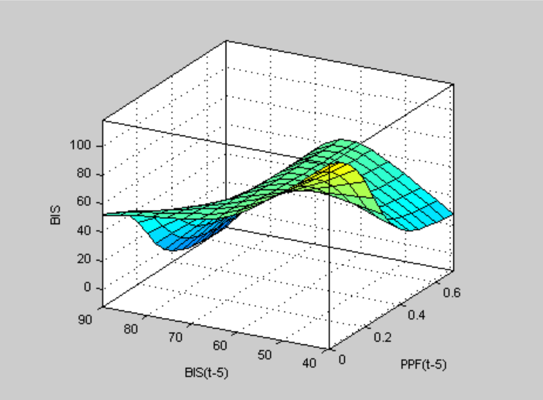


Fig. 4. Surface viewer for system with PPF (t-5) and BIS (t-5) as inputs

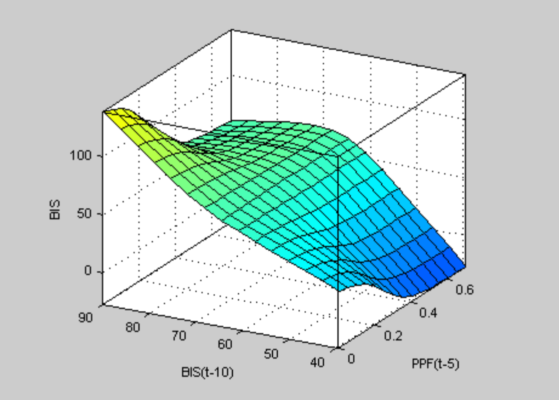


Fig. 5. Surface viewer for system with PPF (t-5) and BIS (t-10) as inputs

3 Results and Discussion

To validate the model, ANFIS models trained for each patients were tested on the other two patients. The performance of the ANFIS models according to root mean square error (*rmse*) were tabulated in Table 3. It was found that the model can predict the BIS values for other patients very well. Here, the *rmse* was calculated as the square error between filtered BIS and predicted BIS.

Table 3. Performance of ANFIS model according to root mean square error

		Test Data		
		Patient 1	Patient 2	Patient 3
Train Data	Patient 1	0.0223	0.0439	0.0420
	Patient 2	0.0297	0.0205	0.0497
	Patient 3	0.0694	0.1338	0.0339

Table 3 shows that model trained for patient 2 has the best prediction output. This is due to its adequate informative data points with larger range of BIS values which can represent the system sufficiently. The results of BIS prediction for all three patients tested on the ANFIS model trained for patient 2 were shown in figures 6-8.

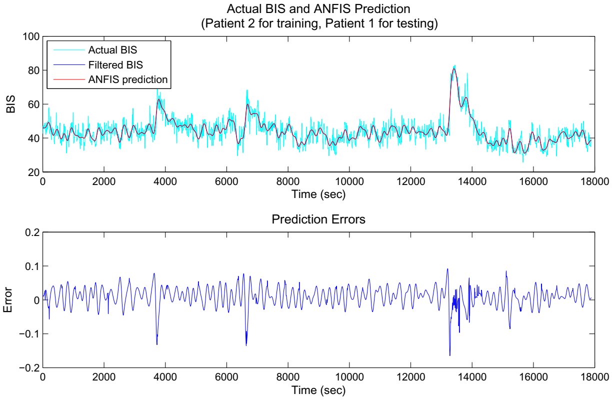


Fig. 6. Predicted BIS output of Patient 1 when applied with model trained for Patient 2

ANFIS model favours conventional model in predicting BIS in a few aspects. First, ANFIS is well suited for predicting uncertain nonlinear system. Furthermore, ANFIS allowed past values of BIS to have influence in predicting future trend of BIS. In fact, sequential search shows that past values of BIS were more significant model inputs than propofol infusion rate. Also, ANFIS can be used for data mining to identify or verify other input variables which will affect BIS.

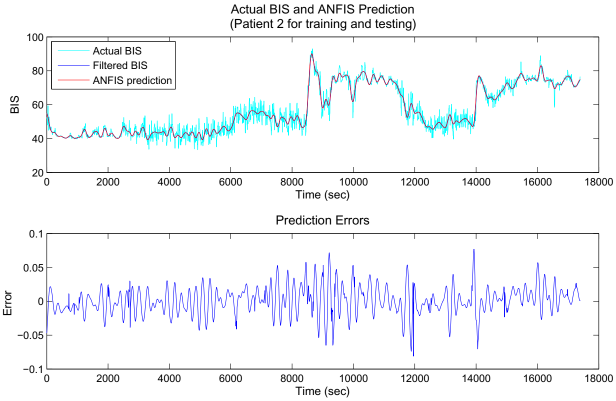


Fig. 7. Predicted BIS output of Patient 2 using model trained for Patient 2

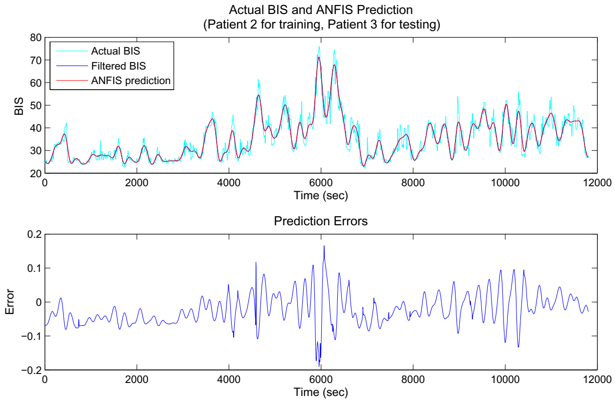


Fig. 8. Predicted BIS output of Patient 3 when applied with model trained for Patient 2

Nevertheless, the ANFIS model will suffer from bad performance if low-quality data set was used for training. The training data set has a direct and decisive effect on the performance of neural-network model [18]. Hence, data preprocessing techniques such as data selection, elimination of noise data, and removal of outliers data are essential. However, no standard method or theory on process of training data set is available.

4 Conclusion

This paper demonstrates the effectiveness of ANFIS model in predicting the BIS using propofol infusion rate and past value of BIS as input variables. However, the present sample size is undoubtedly too small for generalization. Future research efforts will be made to validate the model soon after more data become available.

Acknowledgement. This work is funded by ERGS project number ERGS/1-2012/ 5527093 supported by Ministry Of Higher Education Malaysia.

References

1. Shafer, S., Stanski, D.: Defining depth of anesthesia. In: Schttler, J., Schwilden, H. (eds.) Modern Anesthetics, Handbook of Experimental Pharmacology, vol. 182, pp. 409–423. Springer, Heidelberg (2008)
2. Musizza, B., Ribaric, S.: Monitoring the depth of anaesthesia. Sensors 10(12), 10,896–10,935 (2010)
3. Fahlenkamp, A.V., Peters, D., Biener, I.A., et al.: Evaluation of bispectral index and auditory evoked potentials for hypnotic depth monitoring during balanced xenon anaesthesia compared with sevoflurane. Br. J. Anaesth. 105(3), 334–341 (2010)
4. Rampil, I.J.: A primer for EEG signal processing in anesthesia. Anesthesiology 89(4), 980–1002 (1998)
5. Hahn, J.O., Dumont, G., Ansermino, J.: A direct dynamic dose-response model of propofol for individualized anesthesia care. IEEE Trans. Biomed. 59(2), 571–578 (2012)
6. Rocha, C., Mendona, T., Silva, M.: Individualizing propofol dosage: a multivariate linear model approach. J. Clin. Monitor. Comp., 1–12 (2013)
7. Sartori, V., Schumacher, P.M., Bouillon, T., Luginbuehl, M., Morari, M.: Online estimation of propofol pharmacodynamic parameters. In: 27th Annual International Conference of the Engineering in Medicine and Biology Society, IEEE-EMBS 2005, Shanghai, China, pp. 74–77 (2005)
8. Sawaguchi, Y., Furutani, E., Shirakami, G., Araki, M., Fukuda, K.: A model-predictive hypnosis control system under total intravenous anesthesia. IEEE Trans. Biomed. 55(3), 874–887 (2008)
9. Jang, J.S.: Anfis: adaptive-network-based fuzzy inference system. IEEE Trans. Syst., Man, Cybern., Syst. 23(3), 665–685 (1993)
10. Jang, J.S.: Input selection for anfis learning. In: Proceedings of the Fifth IEEE International Conference on Fuzzy Systems, vol. 2, pp. 1493–1499 (1996)
11. Esmaeili, V., Assareh, A., Shamsollahi, M.B., Moradi, M., Arefian, N.M.: Designing a fuzzy rule based system to estimate depth of anesthesia. In: IEEE Symposium on Computational Intelligence and Data Mining, CIDM 2007, Honolulu, pp. 681–687 (2007)
12. Robert, C., Karasinski, P., Arreto, C., Gaudy, J.: Monitoring anesthesia using neural networks: A survey. J. Clin. Monitor. Comp. 17(3-4), 259–267 (2002)
13. Zhang, X.S., Roy, R.J.: Derived fuzzy knowledge model for estimating the depth of anesthesia. IEEE Trans. Biomed. 48(3), 312–323 (2001)
14. Brás, S., Gouveia, S., Ribeiro, L., Ferreira, D., Antunes, L., Nunes, C.: Fuzzy logic model to describe anesthetic effect and muscular influence on EEG cerebral state index. Res. Vet. Sci. 94(3), 735–742 (2013)
15. Jensen, E., Nebot, A.: Comparison of fir and anfis methodologies for prediction of mean blood pressure and auditory evoked potentials index during anaesthesia. In: Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Hong Kong, vol. 3, pp. 1385–1388 (1998)

16. Baig, M.M., Gholam-Hosseini, H., Lee, S.W., Harrison, M.: Detection and classication of hypovolaemia during anaesthesia. In: 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC, pp. 357–360 (2011)
17. Nunes, C., Amorim, P.: A neuro-fuzzy approach for predicting hemodynamic responses during anesthesia. In: 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS 2008, pp. 5814–5817 (2008)
18. Zhou, Y., Wu, Y.: Analyses on influence of training data set to neural network supervised learning performance. In: Jin, D., Lin, S. (eds.) CSISE 2011. AISC, vol. 106, pp. 19–25. Springer, Heidelberg (2011)

Classify a Protein Domain Using SVM Sigmoid Kernel

Ummi Kalsum Hassan[1], Nazri Mohd Nawi[2], Shahreen Kasim[2], Azizul Azhar Ramli[2], Mohd Farhan Md Fudzee[2], and Mohamad Aizi Salamat[2]

[1] Faculty of Computer Science and Information Technology,
Kolej Poly-Tech MARA, Batu Pahat, Johor, Malaysia
[2] Software Multimedia Centre, Faculty of Computer Science and Information Technology,
Universiti Tun Hussain Onn, Batu Pahat, Johor, Malaysia

Abstract. Protein domains are discrete portion of protein sequence that can fold independently with their own function. Protein domain classification is important for multiple reasons, which including determines the protein function in order to manufacture new protein with new function. However, there are several issues that need to be addressed in protein domain classification which include increasing domain signal and accurate classify to their category. Therefore, to overcome this issue, this paper proposed a new approach to classify protein domain from protein subsequences and protein structure information using SVM sigmoid kernel. The proposed method consists of three phases: Data generating, creating sequence information and classification. The data generating phase selects potential protein and generates clear domain information. The creating sequence information phase used several calculations to generate protein structure information in order to optimize the domain signal. The classification phase involves SVM sigmoid kernel and performance evaluation. The performance of the approach method is evaluated in terms of sensitivity and specificity on single-domain and multiple-domain using dataset SCOP 1.75. The result on SVM sigmoid kernel shown higher accuracy compare with single neural network and double neural network for single and multiple domain prediction. This proposed approach develops in order to solve the problem of coincidently group into both categories either single or multiple domain. This method showed an improvement of classification in term of sensitivity, specificity and accuracy.

Keywords: protein domain, protein sequence, protein structure, support vector machine, protein subsequence.

1 Introduction

Protein sequence consists of protein domain information where domain is a basic unit of protein structure. Protein domains can be seen as distinct functional or structural units of a protein. Protein domains provide one of the most valuable information for the prediction of protein structure, function, evolution and design. The protein sequence may be contained of single-domain or several domains with different or matching copies of protein domain. Each categories of domain may have specific function associated with it. Therefore, classification protein domain into their

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_14, © Springer International Publishing Switzerland 2014

categories such as single-domain or multiple domains are becoming very important in order to understand the protein structure, function and biological processes. However, classifications have a problem itself. In protein domain classification, the protein sequence can exist in more than one category.

Prior methods in classification of protein domains can be classified into five categories. However, these methods produce good results in cases of single-domain proteins. Methods based on similarity and used multiple sequence alignments to represent domain boundaries such as SVM-Fold [1], EVEREST [2] and Biozon [3]. Methods that depend on known protein structure to identify the protein domain such as AutoSCOP [4], Class of Architecture, Topology and Homologous superfamily (CATH) [5] and Structural Classification of Proteins (SCOP) [6]. Methods that used dimensional structure to assume protein domain boundaries such as PROMALS [7], DDBASE [8] and Mateo [9]. Methods that used a model such as Hidden Markov Model (HMM) to identify other member of protein domain family such as Protein Family database (Pfam) [10], Conserved Domain Database (CDD) [11] and Simple Modular Architecture Research Tool (SMART) [12]. Methods that are solely based on protein sequence information such as Domain Guess by Size (DGS) [13] which predict protein domain based on the conformational entropy.

Classification algorithm is a procedure for selecting a hypothesis from a set of alternatives where that is best fits a set of observations. Classification algorithm also does mapping from training sets to hypotheses where that is minimizes the objective function. The goal of classification is to build a set of models that can correctly predict the class of the different objects. The input to classification algorithm is a set of objects which is called as training dataset, the classes which these objects belong to dependent variables and a set of variables describing different characteristics of the objects. Once such a predictive model is built, it can be used to classify the class of the objects for which class information is not known a priori.

Classification of proteins provides valuable clues to structure, activity and metabolic role. Biologist used classification algorithm to organize the function of protein. The function of a protein is directly related to its structure. Basically, the classification is the protein domain rather than whole protein, since the protein domains are the basic unit of evolution, structure and function. The classification algorithm is needed to group the protein into their categories. If the classification algorithm is not applied in biology and bioinformatics, the protein function cannot be determined. Moreover, the new protein also cannot be structured since the protein domain is not classified into their function.

Recently, several classification algorithm have been produced and used in bioinformatics such as algorithms based on fuzzy clustering [14], Neural Network [15], Bayesian classification [16], decision tree [17], logistic regression [18] and Support Vector Machine (SVM) [19, 20]. However, some of these methods were tested on small datasets, often with relatively high sequence identity, which resulted in high classification accuracy such as works done by Chen *et. al.* [21]. The SVM is usually used to map the input vector into one feature space which is relevant with kernel function and seek an optimized linear division that construct the n-separated hyperplane where the n is classes of protein sequence in dataset. These steps are important to make the classification by SVM more accurate and will achieve higher performance.

This paper used the sigmoid kernel function to classify the protein domain into their category. The purpose of this paper is to discover the capability of the modify kernel function in classifying protein domain. Next section will introduce about the approach method beginning from data generation, protein structure information generating and classification by SVM with sigmoid kernel function.

2 Method

Protein domain classifying using modify kernel is tested protein sequence from SCOP version 1.75 [22]. The dataset is split into training and testing datasets. If the protein sequences are longer than 600 amino acids, the protein sequences are separated into a segment based on ordered and disordered regions [23]. Multiple sequence alignment (MSA) is performed in order to give the information of protein domain using Clustal Omega [24]. Then, extract the information to make the protein domain boundaries clearer. This extraction of MSA contains evolutionary information. Then, the information of protein structures is extracted using several measures, and then SVM with modify kernel is applied to classify the protein domain. Lastly, the results from classification of SVM will be evaluated. The proposed approach is shown in Fig. 1.

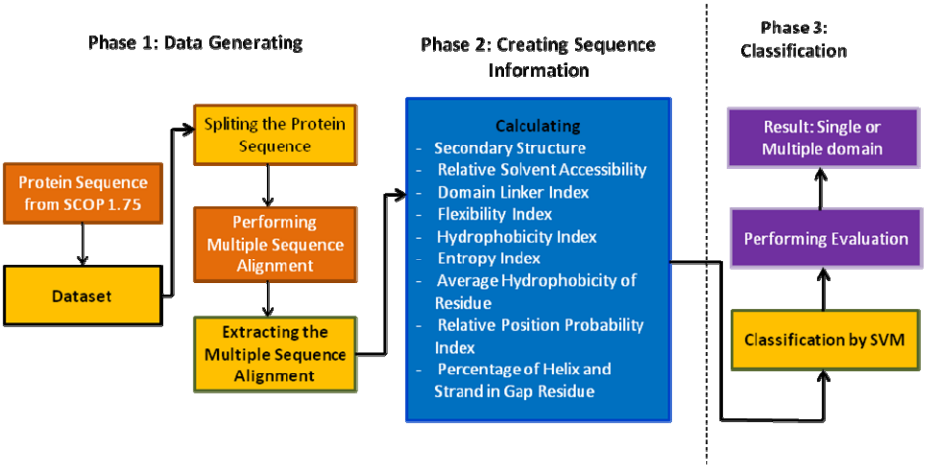

Phase 1: Data Generating
Phase 2: Creating Sequence Information
Phase 3: Classification
Protein Sequence from SCOP 1.75
Dataset
Spliting the Protein Sequence
Performing Multiple Sequence Alignment
Extracting the Multiple Sequence Alignment
Calculating
- Secondary Structure
- Relative Solvent Accessibility
- Domain Linker Index
- Flexibility Index
- Hydrophobicity Index
- Entropy Index
- Average Hydrophobicity of Residue
- Relative Position Probability Index
- Percentage of Helix and Strand in Gap Residue
Classification by SVM
Performing Evaluation
Result: Single or Multiple domain

Fig. 1. Proposed approach

2.1 Dataset

The SCOP 1.75 with 40% less identity in PDB contains 1070. The protein sequences are reconstructed from which short protein sequences that are less than 40 amino acids are removed. The protein sequences that have more than 20 hits using BLAST search are kept.

We divided the dataset into training and testing datasets. Training dataset is used for optimizing the SVM parameters and for training the SVM classifier to predict unseen protein domain. Testing dataset is used for evaluating the performance of the

SVM. The dataset are split into training and testing datasets with 80:20 ratio. Lastly, multiple sequence alignment is performed using Clustal Omega algorithm [24].

Then the pairwise alignments generated by Clustal Omega are extracted. In the extraction, a domain signal is defined as a gap which begins at the N or C terminal. The gap with 45 residues or more will remove and the continuous sequence over 45 residues will remain for generating the protein domain signal. The extractions of pairwise alignment are expected to increase PSI-BLAST e-value [25].

2.2 Generating Sequence Information

Protein structure information extraction in this algorithm is important to determine the protein domain information. Several measures are used in order to generate the information such as secondary structure, relative solvent, domain linker index, flexibility index, hydrophobicity index, entropy index and percentage of helix and strand in gap residue. All the measures are to compute the change that a protein sequence position is part of protein domain information. This information from the calculations is believed to reflect the protein structural properties that have informative protein domain structure and the information is used to classify protein domain [23].

2.3 SVM Kernel

SVM is learning machines based on statistical learning technique that trains classifiers based on polynomial functions, radial basic functions, neural networks and splines. The SVM is applied to identify the protein domain boundaries position. The SVM works by: (1) mapping the input vector into one feature space which is relevant with kernel function; (2) seeking an optimized linear division where that is constructed the *n*-separated hyperplane where the *n* is classes of protein sequence in dataset. The input vector is defined as follows:

$$l_i \in \{+1, -1\} \tag{1}$$

where I_s is the input space with corresponding labels is defined as follows:

$$y_i \in I_s (i = 1, ..., n) \tag{2}$$

where +1 and -1 are used to stand, respectively, for the two classes. The SVM is trained with sigmoid kernel that is used in pattern recognition. The parameter of SVM training is σ as sigmoid kernel smoothing parameter and the SVM training error to generate performance of trade-off parameter. The kernel is defined as follows:

$$K(\vec{y}_i, \vec{y}_j) = \tanh(< \vec{y}_i, \vec{y}_j > + \sigma) \tag{3}$$

where $\vec{y}_i$ is labels and $\vec{y}_j$ is input vector. The input vector will be the centre of the sigmoid kernel and σ will determine the area of influence this input vector has over the data space. A larger value of σ will give a smoother decision surface and a more regular decision boundary since the SVM sigmoid kernel with large σ will allow an input vector to have a strong influence over a larger area. Two parameter are required for optimizing the SVM sigmoid kernel classifier where σ determines the capacity of the sigmoid kernel and the regularization parameter *C*.

The best pair of parameter of *C* and σ is search using *k*-fold cross-validation to evaluate the unbiased data. In this study, *k=10* is applied where the protein sequence into *k* subsets of approximately equal size for integrity of protein sequence. The best combinations_of *C* and σ obtained from the optimization process were used for training the SVM classifier using the entire training dataset. The SVM classifier is subsequently used to predict the testing datasets. The SVM classified the protein domain into single-domain or multiple-domain. Various quantitative variables were obtained to measure the effectiveness of the SVM classification: true positives (TP) for the number of correctly classified as protein domain; false positives (FP) for the number of incorrectly classified as not protein domain; true negatives (TN) for the number of correctly classified as not protein domain; and false negatives (FN) for the number of incorrectly classified as protein domain.

2.4 Evaluation

Using the information from output by SVM, a series of statistical metrics were computed to measure the effectiveness of the algorithm. Sensitivity (SN) and Specificity (SP), which indicates the ability of the prediction system to correctly classify the protein domain and not protein domain respectively, the SN and SP are defined as follows:

$$SN = \left(\frac{(TP)}{(TP + FN} \right) * 100 \tag{4}$$

$$SP = \left(\frac{(TP)}{(TP + FP} \right) * 100 \tag{5}$$

To provide an indication of the overall performance of the system, we computed Accuracy (AC), for the percentage of the correctly predicted protein domain. The AC is defined as follows:

$$AC = \left(\frac{(TP - TN)}{(TP + FN = TN = FP} \right) * 100 \tag{6}$$

3 Results and Discussion

In this study, we test SVM with sigmoid kernel and compare its performance with Single Neural Network and Double Neural Network [26]. The structures protein information are generates from splitting protein sequence. The protein structure gives the strong signal of protein domain boundaries and is used to classify protein domain into their classes either single or multiple domain. Firstly, the SVM sigmoid kernel starts by searching large protein sequences and comparing them with the PDP database to generate multiple sequence alignments and extract the multiple sequence alignment to generate clear signal of domain boundaries. Secondly, calculate the protein structure information from the domain signal information to optimizing the domain signal using several calculation such as secondary structure, relative solvent accessibility, domain linker, flexibility, hydrophobicity, entropy and helix and strand. This score from several calculation used as input to the SVM sigmoid kernel. Then, classify protein domain using SVM with sigmoid kernel. Finally, the results generated by SVM are evaluated. This evaluation provides a clear understanding of strengths and weaknesses of an algorithm that has been designed.

The datasets obtained from SCOP 1.75 version are used to test and evaluate the SVM sigmoid kernel and other classifier done by Kalsum *et. al.*[26]. The results of the accuracy classification compared with other classifier methods including sensitivity and specificity for single-domain or multiple-domain are presented in Table 1 and Fig. 2. It is easy to see that classification of multiple-domain is more difficult than single-domain.

Table 1. Sensitivity and specificity for single and multiple domain prediction

	Single Domain			Multiple Domain		
	SN	SP	ACC	SN	SP	ACC
Single Neural Network	0.68	0.73	0.71	0.76	0.71	0.73
Double Neural Network	0.78	0.87	0.85	0.79	0.82	0.81
SVM Sigmoid Kernel	0.82	0.93	0.9	0.86	0.94	0.96

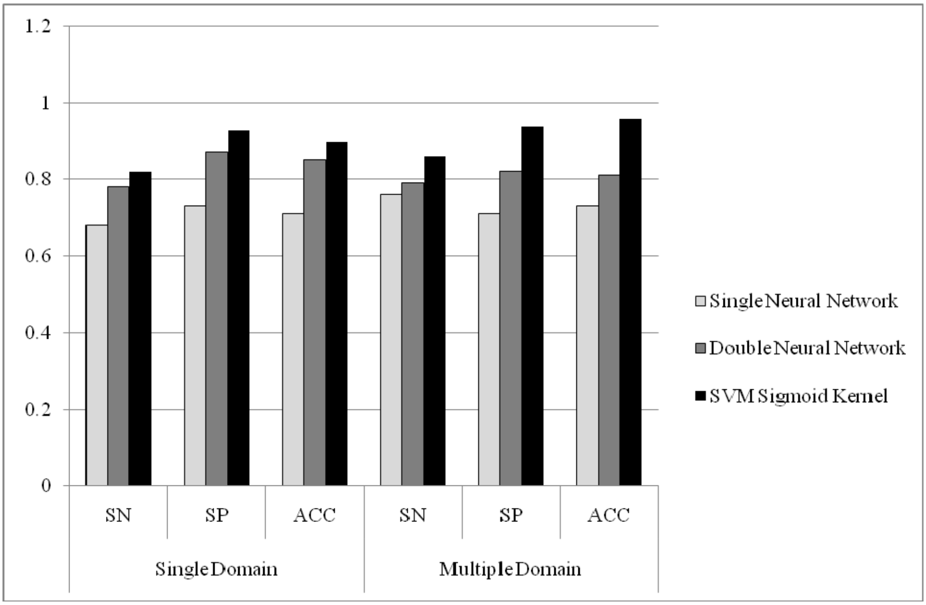


Fig. 2. Sensitivity and specificity chart for single and multiple domain prediction

The SVM sigmoid kernel achieved a higher sensitivity of 82% for single-domain and 86% for the multiple-domain compared to others classifier. The SVM sigmoid kernel achieved a higher specificity of 93% for single-domain and 94% for the multiple-domain compared to others classifier. The SVM sigmoid kernel produces better result of 90% accuracy for single-domain and 96% accuracy for multiple-domain compared with others classifier. The SVM sigmoid kernel are believed to increase the accuracy of classification since sigmoid kernel are used in polynomial case. In this study, the protein sequence may included in two categories either single or multiple domain. The SVM sigmoid kernel makes protein sequence classified clearly and accurate to their category.

4 Conclusions

This proposed approach develops in order to solve the problem of protein sequence can stay in both category at the same time. The algorithm begins with searching the seed protein sequences as dataset from SCOP 1.75. The dataset is split into training and testing sets as ratio 80:20. Then, applied split technique to protein sequence and performed multiple sequence alignment (MSA). After that, extract the MSA to remove gap data. The next step, generate protein sequence information using several measures such as Secondary Structure, Relative Solvent Accessibility, Domain Linker Index, Flexibility Index, Hydrophobicity Index, Entropy Index, Average Hydrophobicity of Residue, Relative Position Probability Index and Percentage of Helix and Strand in Gap Residue to increase signal of protein domain by generating the protein structure. Then, SVM sigmoid kernel is applied to classify the protein domain into single and multiple domain. Lastly, the result from proposed approach evaluated in term of sensitivity and specificity and compare with Single Neural Network and Double Neural Network evaluation. The proposed approach has shown the improvement of prediction with 90% accuracy in single domain classification and 96% accuracy in multiple domain classification compare with Single Neural Network and Double Neural Network. Therefore, the increasing information of protein sequence information is believed increase the domain signal and sequence structure while the using of sigmoid kernel causes for accurate classification.

Acknowledgment. This project is funded by the Universiti Tun Hussein Onn Malaysia under Research Acculturation Grant Scheme (RAGS) vote no. R0001. The comments and suggestions from the anonymous reviewers greatly improved the paper.

References

1. Melvin, I., Weston, J., Leslie, C.S., Noble, W.S.: Combining Classifiers for Improved Classification of Proteins from Sequence or Structure. BMC Bioinformatics 9, 38–389 (2008)
2. Portugaly, E., Harel, A., Linial, N., Linial, M.: EVEREST: Automatic Identification and Classification of Protein Domains in All Protein Sequences. BMC Bioinformatics 7, 27–286 (2006)

3. Nagaranjan, N., Yona, G.: Automatic Prediction of Protein Domain from Sequence Information using a Hybrid Learning System. Bioinformatics 20, 1335–1360 (2004)
4. Gewehr, J.E., Zimmer, R.: SSEP-Domain: Protein Domain Prediction by Alignment of Secondary Structure Elements and Profiles. Bioinformatics 22, 181–187 (2006)
5. Orengo, A., Michie, A.D., Jones, S., Jones, D.T., Swindells, M.B., Thornton, J.M.: CATH-a Hierarchic Classification of Protein Domain Structures. Structure 5, 1093–1108 (1997)
6. Murzin, A.G., Brenner, S.E., Hubbard, T., Chothia, C.: SCOP: A Structural Classification of Protein Database for the Investigation of Sequences and Structures. Journal of Molecular Biology 247, 536–540 (1995)
7. Pei, J., Grishin, N.V.: PROMALS: Towards Accurate Multiple Sequence Alignments of Distantly Related Protein. Bioinformatics 23, 802–808 (2007)
8. Vinayagam, A., Shi, J., Pugalenthi, G., Meenakshi, B., Blundell, T.L., Sowdhamini, R.: DDBASE2.0: Updated Domain Database with Improved Identification of Structural Domains. Bioinformatics 19, 1760–1764 (2003)
9. Lexa, M., Valle, G.: Pimex: Rapid Identification of Oligonucleotide Matches in whole Genomes. Bioinformatics 19, 2486–2488 (2003)
10. Finn, R.D., Mistry, J., Schuster-Bockler, B., Griffiths-Jones, S., Hollich, V., Lassmann, T., Moxon, S., Marshall, M., Khanna, A., Durbin, R., Eddy, S.R., Sonnhammer, E.L., Bateman, A.: Pfam: Clans, Web Tools and Services. Nucleic Acids Research 34, D247–D251 (2006)
11. Marchler, A., Anderson, J.B., Derbyshire, M.K., DeWeese-Scott, C., Gonzales, N.R., Gwadz, M., Hao, L., He, S., Hurwitz, D.I., Jackson, J.D., Zhaoxi, K., Krylov, D., Lanczycki, C.J., Liebert, C.A., Liu, C., Lu, F., Lu, S., Marchler, G.H., Mullokandov, M., Song, J.S., Thanki, N., Yamashita, R.A., Yin, J.J., Zhang, D., Bryant, S.H.: CDD: A Conserved Domain Database for Interactive Domain Family Analysis. Nucleic Acids Research 35, D237–D240 (2005)
12. Letunic, I., Copley, R.R., Pils, B., Pinkert, S., Schultz, J., Bork, P.: SMART 5: Domains in the Context of Genomes and Networks. Nucleic Acids Research 34, D257–D260 (2006)
13. Wheelan, S.J., Marchler-Bauer, A., Bryant, S.H.: Domain Size Distributions can Predict Domain Boundaries. Bioinformatics 16, 613–618 (2000)
14. Lu, T., Dou, Y., Zhang, C.: Fuzzy clustering of CPP family in plants with evolution and interaction analyses. BMC Bioinformatics 14, S10 (2013)
15. Chen, Y., Xu, J., Yang, B., Zhao, Y., He, W.: A novel method for prediction of protein interaction sites based on integrated RBF neural networks. Comput. Biol. Med. 42, 402–407 (2012)
16. Liang, L., Felgner, P.L.: Predicting antigenicity of proteins in a bacterial proteome; a protein microarray and naive Bayes classification approach. Chem. Biodivers. 9, 977–990 (2012)
17. Medina, F., Aguila, S., Baratto, M.C., Martorana, A., Basosi, R., Alderete, J.B., Vazquez-Duhalt, R.: Prediction model based on decision tree analysis for laccase mediators. Enzyme Microb. Technol. 52, 68–76 (2013)
18. Sun, H., Wang, S.: Penalized logistic regression for high-dimensional DNA methylation data with case-control studies. Bioinformatics 28, 1368–1375 (2012)
19. Xin, M., Jiansheng, W., Xiaoyun, X.: Identification of DNA-Binding Proteins Using Support Vector Machine with Sequence Information. Computational Mathematical Methods in Medicine 1, 524502 (2013)
20. Vinay, N., Monalisa, D., Sowmya, S.M., Ramya, K.S., Valadi, K.J.: Identification of Penicillin-binding proteins employing support vector machines and random forest. Bioinformation 9, 481–484 (2013)

21. Ruoying, C., Wenjing, C., Sixiao, Y., Di, W., Yong, W., Yingjie, T., Yong, S.: Rigorous assessment and integration of the sequence and structure based features to predict hot spots. BMC Bioinformatics 12, 311 (2011)
22. David, A., Hai, F., Owen, J.L., Rackham, D.W., Ralph, P., Cyrus, C., Julian, G.: SUPERFAMILY 1.75 including a domain-centric gene ontology method. Nucleic Acids Res. 39, D427–D434 (2011)
23. Kalsum, H.U., Shah, Z.A., Othman, R.M., Hassan, R., Rahim, S.M., Asmuni, H., Taliba, J., Zakaria, Z.: SPlitSSI-SVM: an algorithm to reduce the misleading and increase the strength of domain signal. Comput. Biol. Med. 39, 1013–1019 (2009)
24. Sievers, F., Wilm, A., Dineen, D., Gibson, T.J., Karplus, K., Li, W., Lopez, R., McWilliam, H., Remmert, M., Soding, J., Thompson, J.D., Higgins, D.G.: Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. Mol. Syst. Biol. 7, 539 (2011)
25. Eickholt, J., Deng, X., Cheng, J.: DoBo: Protein domain boundary prediction by integrating evolutionary signals and machine learning. BMC Bioinformatics 12, 1471 (2011)
26. Kalsum, H.U., Nazri, M.N., Shahreen, K.: A New Approach for Protein Domain Prediction by Using Double Stage Neural Network. Adv. Sci. Eng. Med. 6, 129–132 (2014)

Color Histogram and First Order Statistics for Content Based Image Retrieval

Muhammad Imran[1,*], Rathiah Hashim[1], and Noor Eliza Abd Khalid[2]

[1] FSKTM, University Tun Hussein onn Malaysia
86400 Parit Raja, Batu Pahat, Johor, Malaysia
malikimran110@gmail.com, radhiah@uthm.edu.my
[2] FSKTM, University Teknologi Mara Malaysia
Shah Alam, Selangor, Malaysia
elaiza@tmsk.uitm.edu.my

Abstract. Content Based Image Retrieval (CBIR) is one of the fastest growing research areas in the domain of multimedia. Due to the increase in digital contents these days, users are experiencing difficulties in searching for specific images in their databases. This paper proposed a new effective and efficient image retrieval technique based on color histogram using Hue-Saturation-Value (HSV) and First Order Statistics (FOS), namely HSV-*fos*. FOS is used for the extraction of texture features while color histogram deals with color information of the image. Performance of the proposed technique is compared with the Variance Segment and Histogram based techniques and results shows that HSV-*fos* technique achieved 15% higher accuracy as compared to Variance Segment and Histogram-based techniques. The proposed technique can help the forensic department for identification of suspects.

Keywords: Content Based Image Retrieval, First Order Statistics, Color Histogram.

1 Introduction

Retrieving similar image from an image database is a challenging task.There are two approaches available to retrieve an image from an image database. First, retrieval is done based on text and the other uses image content for searching similar images. In the text based image retrieval, traditional database techniques are used to manage images. For searching any specific image from a large database, text based retrieval is not easy because it is highly dependent on keywords and also, the same image can convey different meanings for different people. Text based image retrieval is also called as Keyword Based Image Retrieval (KBIR). Searching images based on the content of the image is called Content Based Image Retrieval (CBIR) which uses different features of the image to search similar images from an image database [1]. The CBIR is suitable for both large and small size databases. By analyzing the image content, we aim at developing new robust techniques for

[*] Corresponding author.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_15, © Springer International Publishing Switzerland 2014

retrieving similar images. The exponentially increasing digital content databases motivate researchers to propose the improved CBIR technique.

Through CBIR, users can search for images using their semantic content such as object categories [2]. The object categories can be forest,vehicle, human, buildings and the like. CBIR adopts feature extraction and selection approaches for image retrieval process. Generally, three types of low level features (texture, color and shape) are extracted from the image to perform the searching process. The retrieval process is performed in three steps. Firstly, the feature which describes the content of the images are extracted. Secondly, extracted features of the image database and the feature vector of the query image are used to find the distance between them. In short, in the second step, the similarity measure is performed between the query image and the image database. Thirdly, the results are indexed and displayed to the user. Therefore, development of effective and efficient feature extraction is a critical part of CBIR systems to achieve better results. The extracted features should represent the image in a way that complies with the human perception subjectivity. Different users analyze images differently, even a user may analyze the same images differently in different situations. That is why, difficult job for CBIR is to understand what user really wants. For better performance of CBIR, researchers proposed a variety of solutions, few of them used local features while others used global features. To extract the local features, image is segmented into regions based on color and texture. Many machine learning techniques are also applied in CBIR system to improve the performance of CBIR. This paper proposed a new signature development technique for the CBIR using color histogram and First Order Statistics (FOS). Rest of the paper is organized as: an overview of related work is presented, section 3 discussed about the proposed approach, results are discusses in section 4 and finally, the paper is concluded in section 5.

2 Related Works

Rao et al.[3] used texture features for image indexing and retrieval. They used the wavelet transformation to construct a feature vector. Euclidean distance was used to measure the image similarity. Authors used the Haar wavelet transform for signature development. To perform clustering, they modified and implemented the ROCK clustering algorithm. The wavelet signature (texture feature representation) is computed from sub image as follows:

$$f_r = \sqrt{\frac{\sum c_{ij}^2}{i \times j}} \tag{1}$$

Where f_r is the computed wavelet signature (texture feature representation) of the sub image, C_{ij} is the representation of the intensity value of all elements of sub image and ij is the size of the sub image. The proposed technique was applied on some garment images and shown better results. Abubacker and Indumathi [4] used color, texture and shape features for the image. To extract color feature they used the spatial based color moments. First they divide the image into 25 blocks then calculate the Red Green Blue (RGB) values of each block. RGB values are converted to Hue, Saturation Intensity (HSI). According to the author, three color moments; mean, variance and skewness are effective and efficient for the color distribution of images. The formula for mean, variance and skewness are given as shown below:

$$Mean(\mu_i) = \frac{1}{N}\sum_{j=1}^{N} f_{ij} \tag{2}$$

$$Varience(\sigma_i) = (\frac{1}{N}\sum_{j=1}^{N}(f_{ij} - \mu_i)^2)^{\frac{1}{2}} \tag{3}$$

$$Skewness(\S_i) = (\frac{1}{N}\sum_{j=1}^{N}(f_{ij} - \mu_i)^3)^{\frac{1}{3}} \tag{4}$$

where f_i is the value of the i^{th} color component of the image block j, and N is the number of blocks in the image. Authors applied 2D Gabor function to obtain the set of Gabor filters with different scale and orientation as a texture feature. Invariant shape features were used to extract the shape features. For extracting the shape feature, following process was used.

1. Based on the threshold value, the image was converted to binary image.
2. Using canny algorithm, the edges of the binary image were detected.
3. The centroid of the object was obtained by arranging the pixels in clockwise order and forming a closed polygon.
4. The centroid distance and complex coordinate function of the edges were found.
5. The farthest points were found and Fourier transform was applied on them.

Imran et al. [5] used color histogram for searching the similar images. Center moment was adopted to describe the histogram. Each image was divided into 4x4 sub images and each sub images is divided into HSV components to generate the histogram. Experiments are performed on Coral Database using precision as performance metric. Broilo and Natale [6] used the Particle Swarm Optimization (PSO) as a classifier to improve the performance of CBIR. PSO was proposed by Kennedy and Eberhart [7] in 1995 and modified by different research works such as [8][9][10]. Huang et al. [11] proposed the new technique of the CBIR using color moment and Gabor texture feature. To obtain the color moment, they converted the RGB image to HSV image. Then, by getting the equalized histogram of the three HSV components, three moments for each color space were calculated. Modified form of the Euclidean distance was used to measure the similarity between the query image and the database image. The equation is given below;

$$D(q,s) = \frac{1}{L}\sum_{i=0}^{L-1}(1 - \frac{|q_i - s_i|}{max(q_i, s_i)}) \quad (5)$$

The global distance is computed as the weighted sum of similarities as

$$D(q,s) = \frac{\omega_c.D_c(q,s) + \omega_t.D_t}{\omega_i + \omega_t} \quad (6)$$

3 Proposed Approach (HSV-*fos*)

This paper proposed a new signature development technique to improve the performance of CBIR. Color and texture feature are extracted and combined to represent an image. The detail of feature extraction is discussed in following subsections.

3.1 Color Histogram

Color feature has an important role in the field of CBIR. It is most sensitive and perceptible feature of the image. To demonstrate color information, different techniques are available. One of them is color histogram. Color histogram is fast and it is not sensitive to change in images such as translation, rotation and the like.

In this paper, center moment is used to illustrate the histogram, the center moment and mean definition as defined by Rafael et.al. [12] are given as:

$$\mu_n = \sum_{i=0}^{L-1}(Z_i - m)^n p(Z_i) \quad (7)$$

$$m = \sum_{i=0}^{L-1} Z_i p(Z_i) \quad (8)$$

where n represent the moment order, Z_i is the gray level, $p(Z_i)$ is normalized histogram, m is mean of histogram and L is the total number of gray level. HSV color Model is used to extract the color feature vector. Following steps define the process of feature extraction:

1. divide the image space into 4x4 sub images which result into 16 sub images as shown in Fig 1.
2. convert each sub image into the HSV image.
3. extract the three components (i.e. H,S,V) from the sub image.
4. for the three components of each sub image, generate the equalized histogram (Histogram equalization is a technique of contrast adjustment through image histogram).
5. compute the three moments (mean, skewness and variance) for each H, S and V histogram.
6. combine the values of all moments in a single feature vector.
7. combine all the 16 feature vectors in a single feature vector.

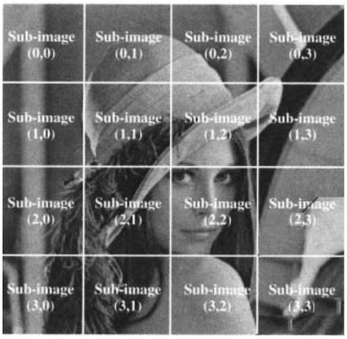


Fig. 1. Definition of sub image

3.2 First Order Statistics

To extract the texture features, first-order histogram is used. First Order texture measures are calculated using original image values. When an image is represented as a histogram, the intensity value concentration on all or part of the histogram is used for the texture analysis. Different features can be derived from the histogram which include moments such as variance, mean, skewness, entropy, Kurtosis and Energy etc [13]. The following equations are used to calculate these moments.

$$Mean : \mu = \sum_{i=1}^{L} k_i p\left(k_i\right) \tag{9}$$

$$Variance : \sigma^2 = \sum_{i=1}^{L} \left(k_i - \mu\right)^2 p\left(k_i\right) \tag{10}$$

$$Skewness : \mu_3 = \sigma^{-3} \sum_{i=1}^{L} \left(k_i - \mu\right)^{-3} p\left(k_i\right) \tag{11}$$

$$Kurtosis : \mu_4 = \sigma^{-4} \sum_{i=1}^{L} \left(k_i - \mu\right)^{-4} p\left(k_i\right) - 3 \tag{12}$$

$$Energy : E = \sum_{i=1}^{L} \left[p\left(k_i\right)\right]^2 \tag{13}$$

$$Entropy : H = -\sum_{i=1}^{L} \left(k_i\right) log_2 \left[p\left(k_i\right)\right] \tag{14}$$

where k_i = gray value of the i^{th} pixel L = number of distinctive gray levels $p(k_i)$= normalized histogram. According to Materka and Strzelecki [14], image intensity is usually represented by a mean value. Intensity variation around the mean is defined as variance. The skewness becomes zero if the histogram is balanced with the mean. When histogram is skewed above the mean, the skewness becomes positive and it becomes negative if histogram is skewed below the mean. The flatness of the histogram is measured as kurtosis. Histogram uniformity is measured by the entropy. The entropy is a measure of histogram consistency.

To calculate the texture signature, variance, mean skewness, energy, entropy and Kurtosis are computed using the above equations. Colri and texture features of the entire image database are extracted, combined and saved to the disk. When user input the query image, the system computes the resemblance between query image and the database image by using Manhattan distance. After similarity measure, the system sorts the result and displays it to the user.

3.3 Experimental Setup

To validate the performance of the proposed system, extensive experiments were performed on real data set. The generated results were compared with other well known CBIR algorithms. We have used the Coral Dataset having 10 classes and each class contains 100 images. 10 images were randomly selected as a query image from each class and perform retrieval process. The process was repeated for 10 times and category wise average precision was computed.

4 Results and Analysis

Results of the proposed system were compared with previous proposed CBIR system namely Variance Segment Method (VSM) [15] and Histogram based (SH) [16]. The results of SH were taken from the work of Banerjee et.al.[17]. The HSV-*fos* were implemented using Matlab R2010b. To measure the performance of the CBIR system different metrics are available. Precision is one of the metric which has been used in the several previous works such as Hiremath et al. [18], Banerjee et al. [17] and Wang et al [19]. This study also used the precision as performance metric and was calculated as;

$$Precision = \frac{NumberofTruePositive}{NumberofTruePositive + FalsePositive} \qquad (15)$$

where 'Number of True Positive' means the total relevant images retrieved and 'False Positive' means irrelevant images. for example user retrieved total 20 images and 15 are the relevant and 5 are irrelevant than $prisons = \frac{15}{15+5}$, which is equal to .75, it means the accuracy is 75%.

4.1 Performance in Terms of Precision

As illustrated above, precision is used to check the performance of the CBIR system. Table-1 shows the performance of the proposed algorithm with different P @ n evaluation. The precision is calculated using the top 40, 30 and 10 ranked

Table 1. Performance at different n

Class	**n=40**	**n=30**	**n=10**
Africa	.86	.93	.85
Beach	.32	.43	.62
Buildings	.52	.49	.54
Buses	.27	.24	.45
Dinosaurs	.94	.99	1
Elephant	.37	.33	.57
Flower	.27	.31	.44
Horses	.43	.54	.67
Mountains	.14	.18	.21
Food	.18	.20	.28
Avg	**.43**	**.46**	**.56**

results. Top 40, top 30 and top 10 or n=40, n=30 and n=10 means 40, 30 and 10 most similar images respectively.

From Table 1 and the graph shown in Figure 2, it is observed that average precision is based on number of top retrievals. Average precision achieved for top 40 is lower than average precision for top 30.

4.2 Comparison with Previous Methods

Table 2 Illustrates that proposed method has better results than Histogram-based and Variance Segment method. Figure 2 represents the performance of proposed methods at different top retrieval for n = 40, 30, and 10, while figure 3 shows the comparison of the proposed method with VSM [15] and SH [16] and it described that the proposed method has better precision than both VSM and SH.

Table 2. Comparison of proposed method with previous methods

Class	**SH** [16]	**VSM** [15]	**HSV-*fos***
Africa	.30	.13	.93
Beach	.30	.26	.43
Buildings	.25	.11	.49
Buses	.26	.17	.24
Dinosaurs	.90	96	.99
Elephant	.36	.34	.33
Flower	.40	.49	.31
Horses	.38	.20	.54
Mountains	.25	.25	.18
Food	.20	.15	.20
Avg	**.36**	**.306**	**.464**

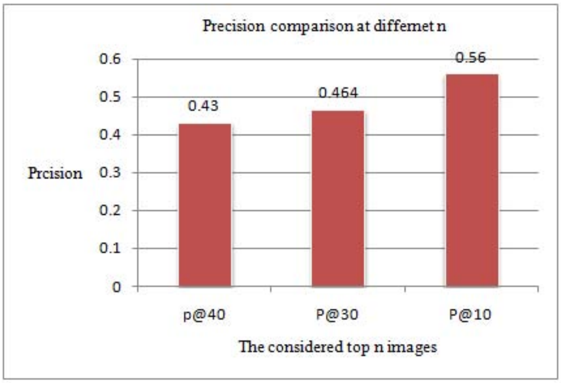


Fig. 2. Comparison of the proposed method with SH and VSM

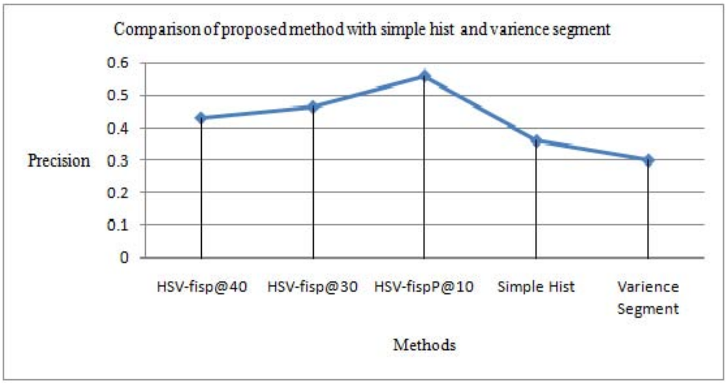


Fig. 3. Comparison of the proposed method with simple Hist and Variance Segment

Figure 4 demonstrates that HSV-*fos* out performed in 8 categories out of 10. Accuracy of proposed technique for class Africa, Beach, Buildings, Buses, Horses and food is 80%, 17%, 11%, 7%, 34% and 5% higher than VSM respectively. For only flower class VSM performed better than HSV-*fos*, both techniques has almost same performance for elephant class.

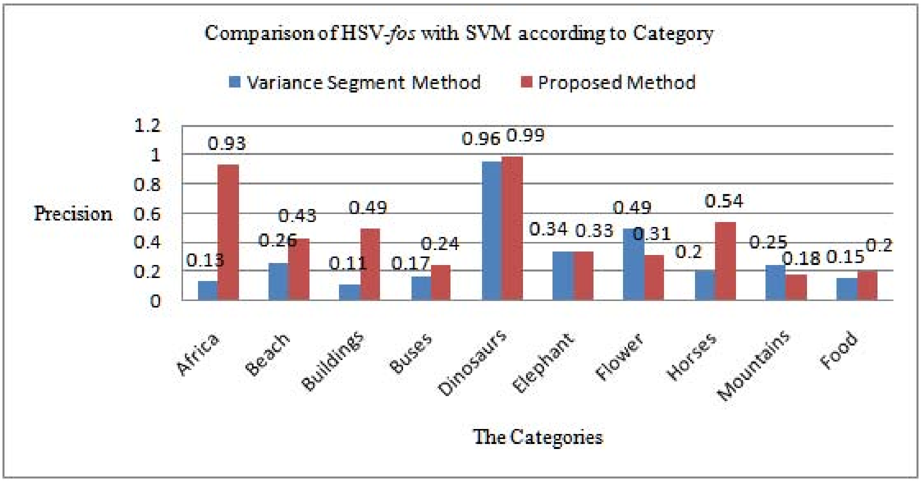


Fig. 4. Category-wise Comparison of Proposed Method (n @ 30) with previous methods

5 Conclusion

Nowadays, a lot of information in the form of digital content are easily accessible on the internet but finding the relevant image is still an issue using current CBIR systems. Hence, this study proposed new signature for CBIR by combining HSV and FOS which is named as HSV-*fos*. The proposed HSV-*fos* system used color and texture features. The color features are extracted by using color histogram while first order statistics are used to extract the texture features. The performance of the system has been evaluated and compared with the previous proposed CBIR techniques. The system is also tested on the different top n image retrieval. During simulation process, it was observed that HSV-*fos* system outperforms, if the 'n' is equal to 30 or less than 30. From the results, it can be clearly seen that the performance of the HSV-*fos* system is better than other CBIR techniques.

Acknowledgment. The researchers would like to thank University Tun Hussein Onn Malaysia (UTHM) for supporting this project under Project Vote No 1315.

References

1. Deshmukh, A., Phadke, G.: An improved content based image retrieval. In: 2nd International Conference on Computer and Communication Technology (ICCCT), pp. 191–195 (September 2011)
2. Sheikh, A., Lye, M., Mansor, S., Fauzi, M., Anuar, F.: A content based image retrieval system for marine life images. In: IEEE 15th International Symposium on Consumer Electronics (ISCE), pp. 29–33 (June 2011)

3. Gnaneswara Rao, K.N., Vijaya, V., Venkata, K.V.: Texture based image indexing and retrieval. IJCSNS International Journal of Computer Science and Network Security 9, 206–210 (2009)
4. Abubacker, K., Indumathi, L.: Attribute associated image retrieval and similarity re ranking. In: International Conference on Communication and Computational Intelligence (INCOCCI), pp. 235–240 (December 2010)
5. Imran, M., Hashim, R., Abd Khalid, N.: New approach to image retrieval based on color histogram. In: Tan, Y., Shi, Y., Mo, H. (eds.) ICSI 2013, Part II. LNCS, vol. 7929, pp. 453–462. Springer, Heidelberg (2013)
6. Broilo, M., De Natale, F.G.B.: A stochastic approach to image retrieval using relevance feedback and particle swarm optimization. IEEE Transactions on Multimedia 12(4), 267–277 (2010)
7. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948 (1995)
8. Imran, M., Hashim, R., Khalid, N.: Opposition based particle swarm optimization with student t mutation (ostpso). In: 4th Conference on Data Mining and Optimization (DMO), pp. 80–85 (2012)
9. Imran, M., Manzoor, Z., Ali, S., Abbas, Q.: Modified particle swarm optimization with student t mutation (stpso). In: International Conference on Computer Networks and Information Technology (ICCNIT), pp. 283–286 (2011)
10. Imran, M., Hashim, R., Khalid, N.E.A.: An overview of particle swarm optimization variants. Procedia Engineering 53, 491–496 (2013)
11. Huang, Z.-C., Chan, P., Ng, W., Yeung, D.: Content-based image retrieval using color moment and gabor texture feature. In: International Conference on Machine Learning and Cybernetics (ICMLC), vol. 2, pp. 719–724 (July 2010)
12. Rafael, R.E.W., Gonzalez, C., Eddins, S.L.: Digital image processing using matlab. Publishing House of Electronics Industry (2009)
13. Selvarajah, S., Kodituwakku, S.R.: Analysis and comparison of texture features for content based image retrieval. International Journal of Latest Trends in Computing 2, 108–113 (2011)
14. Materka, A., Strzelecki, M.: Texture analysis methods a review. Technical University of Lodz, Institute of Electronics (1998)
15. Bhuravarjula, H., Kumar, V.: A novel content based image retrieval using variance color moment. International Journal of computer and Electronic Research 1, 93–99 (2012)
16. Rubner, Y., Guibas, L.J., Tomasi, C.: The earth mover's distance, multidimensional scaling, and color-based image retrieval. In: Proceedings of the ARPA Image Understanding Workshop, pp. 661–668 (1997)
17. Banerjee, M., Kundu, M.K., Maji, P.: Content-based image retrieval using visually significant point features. Fuzzy Sets and Systems 160, 3323–3341 (2009)
18. Hiremath, P., Pujari, J.: Content based image retrieval using color boosted salient points and shape features of an image. International Journal of Image Processing 2(1), 10–17 (2008)
19. Wang, J., Li, J., Wiederhold, G.: Simplicity: semantics-sensitive integrated matching for picture libraries. IEEE Transactions on Pattern Analysis and Machine Intelligence 23, 947–963 (2001)

Comparing Performances of Cuckoo Search Based Neural Networks

Nazri Mohd Nawi[1,2], Abdullah Khan[2], M.Z. Rehman[2],
Tutut Herawan[3,4], and Mustafa Mat Deris[2]

[1] Software and Multimedia Centre (SMC)
[2] Faculty of Computer Science and Information Technology
Universiti Tun Hussein Onn Malaysia
86400, Parit Raja, Batu Pahat, Johor, Malaysia
[3] Department of Information System
University of Malaya
50603 Pantai Valley, Kuala Lumpur, Malaysia
[4] AMCS Research Center, Yogyakarta, Indonesia
{nazri,mmustafa}@uthm.edu.my, hi100010@siswa.uthm.edu.my,
zrehman862060@gmail.com, tutut@um.edu.my

Abstract. Nature inspired meta-heuristic algorithms provide derivative-free solutions to solve complex problems. Cuckoo Search (CS) algorithm is one of the latest additions to the group of nature inspired optimization heuristics. In this paper, Cuckoo Search (CS) is implemented in conjunction with Back propagation Neural Network (BPNN), Recurrent Neural Network (RNN), and Levenberg Marquardt back propagation (LMBP) algorithms to achieve faster convergence rate and to avoid local minima problem. The performances of the proposed Cuckoo Search Back propagation (CSBP), Cuckoo Search Levenberg Marquardt (CSLM) and Cuckoo Search Recurrent Neural Network (CSRNN) algorithms are compared by means of simulations on OR and XOR datasets. The simulation results show that the CSRNN performs better than other algorithms in terms of convergence speed and Mean Squared Error (MSE).

Keywords: Back propagation, Neural network, Cuckoo search, Local minima, Meta-heuristic optimization.

1 Introduction

An Artificial Neural Network (ANN) is a data processing model that emulates the biological nervous systems operations to processes data [1, 2]. ANN consists of a large number of tremendously integrated processing elements (neurons) functioning together to solve many complex real world problems [3]. ANN have been effectively implemented in all engineering fields such as biological modeling, decision and control, health and medicine, engineering and manufacturing, marketing, ocean exploration and so on [4-9]. Because of this complex processing quality, many new

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_16, © Springer International Publishing Switzerland 2014

algorithms have been proposed that inherit the architecture of ANNs in the recent decades. The Back propagation (BP) algorithm is one such architecture of ANN by Rumelhart [10] well-known for training a multilayer feed-forward ANN [11]. However, BP algorithm suffers from two major drawbacks: low convergence rate and instability. They are caused due to getting trapped in a local minimum or due to the overshooting of the minimum of the error surface [12-14]. In recent years, a number of studies have attempted to overcome these problems. They fall into two main categories. The first category uses heuristic techniques developed from an analysis of the performance of the standard steepest descent algorithm. They include the gradient descent with adaptive learning rate, gradient descent with momentum, gradient descent with momentum and adaptive learning rate, and the resilient algorithm. In the standard steepest descent, the learning rate is fixed and its optimal value is always hard to find [11, 12]. The second category uses standard numerical optimization techniques. This includes conjugate gradient [15, 16], quasi-Newton, and Levenberg-Marquardt (LM) algorithm. In the conjugate gradient algorithms, search is performed along conjugate directions. However, one limitation of this procedure, which is a gradient-descent technique, is that it requires a differentiable neuron transfer function. Also, as neural networks generate complex error surfaces with multiple local minima, the BP fall into local minima in place of a global minimum [17, 18]. Many methods have been proposed to speed up the back propagation based training algorithms by fixing the proper learning rate and the momentum value for each layer at the time of training [19]. Different initialization techniques [20, 21] and cost optimization techniques [22], and global search technique such as hybrid PSO-BP [23], Artificial Bee Colony [24-26], Evolutionary algorithms (EA) [27], Particle Swarm Optimization (PSO) [28], Differential Evolution (DE) [29], Ant Colony, and Back propagation [30], Genetic algorithms (GA) [31], have been introduced to intensify the rate of convergence. Cuckoo Search (CS) is a new meta-heuristic search algorithm, developed by Yang and Deb [32] which imitates animal behavior and is valuable for global optimization [33, 34]. The CS algorithm has been applied alone to solve several engineering design optimization problems, such as the design of springs and welded beam structures, and forecasting [33, 35].

In this paper, Cuckoo Search (CS) is implemented in conjunction with Back propagation Neural Network (BPNN), Recurrent Neural Network (RNN), and Levenberg Marquardt back propagation (LMBP) algorithms to achieve faster convergence rate and to avoid local minima problem. The performances of the proposed Cuckoo Search Back propagation (CSBP), Cuckoo Search Levenberg Marquardt (CSLM) and Cuckoo Search Recurrent Neural Network algorithms are compared with other similar hybrid variants based on Mean Squared Error (MSE), CPU time, accuracy and convergence rate to global minima.

The remaining paper is organized as follows: Learning algorithms are presented in the Section 2, while Section 3 deals with the simulation results and discussions and finally the paper is concluded in the Section 4.

2 Learning Algorithms

2.1 Levy Flight

Levy Flights have been used in many search algorithms [37]. In Cuckoo Search algorithm levy flight is an important component for local and global searching [38]. Levy Flights is a random walk that is characterized by a series of straight jumps chosen from a heavy-tailed probability density function [37]. In statistical term, it is a stochastic algorithm for global optimization that finds a global minimum [38]. Each time Levy Flights processes, step length can be calculated by using Mantegna's algorithm as given in the Equation 1.

$$S = \frac{u}{|v|^{\frac{1}{\beta}}} \tag{1}$$

Note that u and v are drawn from normal distribution with respect to these two random variables;

$$u \sim N(0.\sigma_u{}^2), \qquad v \sim N(0.\sigma_v{}^2) \tag{2}$$

The $\sigma_u{}^2$ and $\sigma_v{}^2$ present in Equation 2 are the variance of distributions which come from Equation 3;

$$\sigma_u = \left\{\frac{\Gamma(1+\beta).\sin\,(\tau.\beta/2)}{\Gamma[(1+\beta)/2].\beta.2^{\frac{(\beta-1)}{2}}}\right\}^{\frac{1}{\beta}}, \qquad \sigma_v = 1 \tag{3}$$

The symbol ~ in Equations 2 means the random variable obeys the distribution on right hand side; that is, samples should draw from the distribution.

2.2 Cuckoo Search (CS) Algorithm

Cuckoo Search (CS) algorithm is a new meta-heuristic technique proposed by Xin-She Yang [32]. This algorithm was stimulated by the obligate brood parasitism of some cuckoo species by laying their eggs in the nests of other host birds. Some host nest can keep direct difference. If an egg is discovered by the host bird as not its own, it will either throw the unknown egg away or simply abandon its nest and build a new nest elsewhere. The CS algorithm follows three idealized rules:

a) Each cuckoo lays one egg at a time, and put its egg in randomly chosen nest;
b) The best nests with high quality of eggs will carry over to the next generations;
c) The number of available host nests is fixed, and the egg laid by a cuckoo is discovered by the host bird with a probability $pa \in [0, 1]$.

In this case, the host bird can either throw the egg away or abandon the nest, and build a completely new nest. The Rule (c) defined above can be approximated by the fraction pa $\in [0, 1]$ of the n nests that are replaced by new nests (with new random solutions). When generating new solutions $\mathrm{x}^{\mathrm{t+1}}$ for a cuckoo i, a Levy flight is performed;

$$\mathrm{x_i}^{\mathrm{t+1}} = \mathrm{x_i}^{\mathrm{t}} + \alpha \oplus \mathrm{levy}(\lambda) \ , \tag{4}$$

where $\alpha > 0$ is the step size, which should be related to the scales of the problem of interest. The product ⊕ means entry wise multiplications. The random walk via Levy flight is more efficient in exploring the search space as its step length is much longer in the long run. The Levy flight essentially provides a random walk while the random step length is drawn from a Levy distribution as shown in the Equation 5:

$$\mathrm{Lavy} \sim \mathrm{u} = \mathrm{t}^{-\lambda} \ , 1 < \lambda \leq 3 \tag{5}$$

This has an infinite variance with an infinite mean. Here the steps essentially construct a random walk process with a power-law step-length distribution and a heavy tail.

2.3 Levenberg Marquardt (LM) Algorithm

One reason for selecting a learning algorithm is to speed up convergence. The Levenberg-Marquardt (LM) algorithm is an approximation to Newton's method to accelerate training speed. Benefits of applying LM algorithm over variable learning rate and conjugate gradient method were reported in [39]. The LM algorithm is developed through Newton's method. Assume the error function is:

$$E(t) = \frac{1}{2}\sum_{i=1}^{N} e_i^{\,2}(t) \ , \tag{6}$$

where, $e(t)$ is the error; N is the number of vector elements, then:

$$\nabla E(t) = J^T(t)e(t) \tag{7}$$

$$\nabla^2 E(t) = J^T(t)J(t) + S(t) \ , \tag{8}$$

where, $\nabla E(t)$ is the gradient; $\nabla^2 E(t)$ is the Hessian matrix of E(t);

$$S(t) = \sum_{i=1}^{N} e_i(t)\, \nabla^2 \mathrm{e_i}(t) \tag{9}$$

where $J(t)$ is the Jacobian Matrix;

$$J\,(t) = \begin{bmatrix} \frac{\partial v_1(t)}{\partial t_1} & \frac{\partial v_1(t)}{\partial t_2} \cdots\cdots & \frac{\partial v_1(t)}{\partial t_n} \\ \frac{\partial v_2(t)}{\partial t_1} & \frac{\partial v_2(t)}{\partial t_2} \cdots\cdots & \frac{\partial v_2(t)}{\partial t_n} \\ & \cdot & \\ & \cdot & \\ & \cdot & \\ \frac{\partial v_n(t)}{\partial t_1} & \frac{\partial v_n(t)}{\partial t_2} \cdots\cdots & \frac{\partial v_n(t)}{\partial t_n} \end{bmatrix} \tag{10}$$

For Gauss-Newton Method:

$$\nabla t = -[J^T(t)J(t)]^{-1}J(t)e(t) \tag{11}$$

For the Levenberg-Marquardt algorithm as the variation of Gauss-Newton Method:

$$\nabla t = -[J^T(t)J(t) + \mu I]^{-1}J(t)e(t) \tag{12}$$

where $\mu > 0$ and is a constant; I is identity matrix. So that the algorithm will approach Gauss- Newton, which should provide faster convergence.

3 Results and Discussions

Basically, the main focus of this paper is to compare of different algorithm on based of error, accuracy in network convergence. Before going to discussing the simulation results, there are certain things that need be explained such as tools and technologies, network topologies, testing methodology and the classification problems used for the entire experimentation. The discussion is as follows;

3.1 Preliminary Study

In order to illustrate the performance of the algorithm in training ANN. The simulation experiment is performed on Intel Pentium 3.00 GHz CPU with a 2-GB RAM. The software used for simulation is MATLAB 2008a. In this paper these following algorithms are compared analyzed and simulated on 2-bit XOR, 3-bit XOR and 4-bit OR datasets;

a) Cuckoo Search Levenberg Marquardt (CSLM) algorithm [40],
b) Cuckoo Search Back propagation (CSBP) algorithm [41],
c) Cuckoo Search Recurrent neural network (CSRNN) [42],

The performance measure for each algorithm is based on the Mean Square Error (MSE), standard deviation and accuracy. The three layers feed forward neural network architecture (i.e. input layer, one hidden layer, and output layers.) is used for each problem. The number of hidden nodes is keep fixed to 5. In the network structure, the bias nodes are also used and the log-sigmoid activation function is applied. For each problem, trial is limited to 1000 epochs. A total of 20 trials are run for each dataset. The network results are stored in the result file for each trial. CPU time, average accuracy, and Mean Square Error (MSE) are recorded for each independent trial on XOR and OR datasets.

3.2 2-Bit XOR Dataset

The first test problem is the 2 bit XOR Boolean function of two binary inputs and a single binary output. In the simulations, we used 2-5-1, feed forward neural network for two bit XOR dataset. The parameters range for the upper and lower band is set to [5,-5] respectively, for the CSLM, CSBP, and CSRNN algorithm. Table 1 shows the CPU time, number of epochs, MSE, Standard Deviation (SD), and Accuracy for the 2 bit XOR test problem with 5 hidden neurons. Figure1 shows the 'MSE performances vs. Epochs' of CSLM, CSBP, and CSRNN algorithms for the 2-5-1 network structure. Although, CSLM, CSBP, and CSRNN performed well on 2-Bit XOR but the convergence rate of CSRNN was better. The CSRNN converged in 7 epochs and took 0.78 CPU seconds. Figure 1 shows the graphical representation of the MSE, SD, Epoch, and Accuracy. From Figure 1, it's conformed that CSRNN is better in terms of convergence and CPU time than the CSBP and CSLM algorithms.

Table 1. CPU Time, Epochs, MSE, Accuracy, Standard Deviation (SD) for 2-5-1 ANN Architecture

Algorithm	CSRNN	CSBP	CSLM
CPU Time	0.78	21.22	15.80
Epochs	7	134	145
MSE	0	0	0
SD	0	0	0
Accuracy (%)	100	100	100

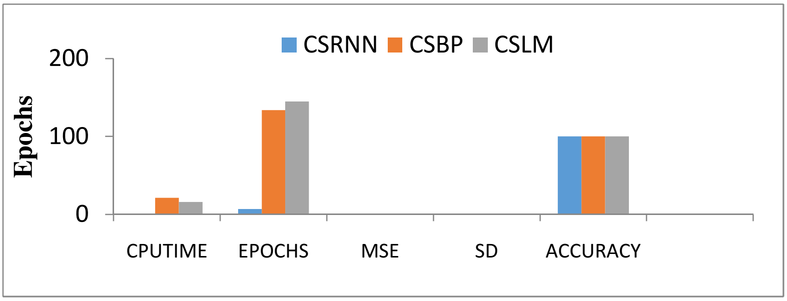


Fig. 1. Performance Comparison of CSRNN, CSLM, and CSBP algorithms on the basis of CPU Time, Epochs, MSE, Accuracy, Standard Deviation (SD) for 2-5-1 ANN Architecture

3.3 3-Bit XOR Dataset

In the second phase, we used 3 bit XOR dataset consisting of three inputs and a single binary output. For the three bit input we apply 3-5-1, network architecture. The parameter range is same as used for two bit XOR problem, for the 3-5-1 the network has twenty connection weights and six biases. Table 2 shows the CPU time, number of epochs, the MSE standard deviation (SD) and accuracy for CSRNN, CSBP, and CSLM algorithms. From Table 2, it is clearly visible that CSRNN converged with a superior 0.82 CPU cycles, 8 epochs and an MSE of 0 with 100 percent accuracy while CSBP and CSLM follows behind. Figure 2 illustrates the performance of the proposed algorithms.

Table 2. CPU Time, Epochs, MSE, Accuracy, Standard Deviation (SD) for 3-5-1 ANN Architecture

Algorithm	**CSRNN**	**CSBP**	**CSLM**
CPU Time	0.82	149.53	80.36
Epochs	8	938	671
MSE	0	5.4E-04	7.5E-07
SD	0	0.00134	3.14E-06
Accuracy (%)	100	98.7351	99.99

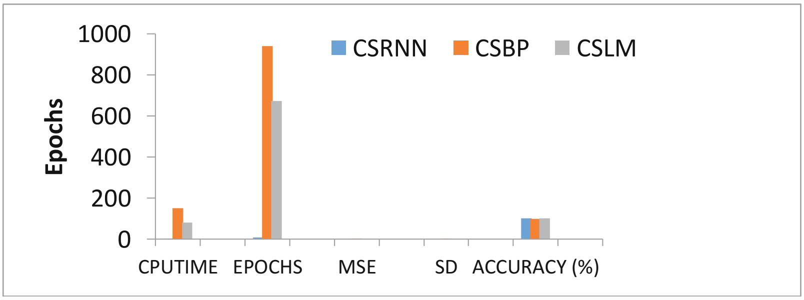


Fig. 2. Performance Comparison of CSRNN, CSLM, and CSBP algorithms on the basis of CPU Time, Epochs, MSE, Accuracy, Standard Deviation (SD) for 3-5-1 ANN Architecture

3.4 4-Bit OR Dataset

The third dataset is based on the logical operator OR which indicates whether an operand is true or false. If one of the operand has a nonzero value the result has a value equal to 1, otherwise it has a 0 value. The network architecture used here is 4-5-1 in which the network has twenty five connection weights and six biases. Table 3, illustrates the CPU time, epochs, and MSE performance and accuracy of the used algorithms, such as CSRNN, CSBP, and CSLM algorithms respectively. Figure 3, shows the graphical representation for the 4-5-1 network architecture of CSRNN, CSBP, and CSLM algorithms. In Figure 3, we can see that the hybrid Cuckoo Search algorithms achieved 0 MSE with 100 percent accuracy. The simulation results show that the CSRNN has better performance than others algorithms in terms of epochs, and CPU time.

Table 3. CPU Time, Epochs, MSE, Accuracy, Standard Deviation (SD) for 4-5-1 ANN Architecture

Algorithm	**CSRNN**	**CSBP**	**CSLM**
CPU Time	1.83	8.48	6.16
Epochs	13	51	55
MSE	0	0	0
SD	0	0	0
Accuracy (%)	100	100	100

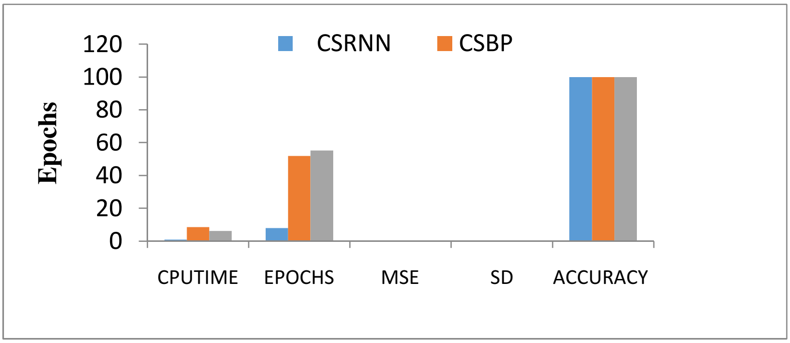


Fig. 3. Performance Comparison of CSRNN, CSLM, and CSBP algorithms on the basis of CPU Time, Epochs, MSE, Accuracy, Standard Deviation (SD) for 4-5-1 ANN Architecture

4 Conclusions

Nature inspired meta-heuristic algorithms provide derivative free solution to optimize complex problems. A new meta-heuristic search algorithm, called Cuckoo Search (CS) is an optimization algorithm developed by Xin-She Yang [32]. In this paper, CS is incorporated with Back propagation Neural Network (BPNN), Recurrent Neural Network (RNN), and Levenberg Marquardt back propagation (LMBP) algorithms to achieve faster convergence rate and to avoid local minima problem. The performances of the proposed Cuckoo Search Back propagation (CSBP), Cuckoo Search Levenberg Marquardt (CSLM) and Cuckoo Search Recurrent Neural Network (CSRNN) algorithms are verified by means of simulation on three datasets such as 2-bit, 3-bit XOR and 4-bit OR. The simulation results show that the proposed CSRNN algorithm is far better than the CSBP and CSLM algorithms in terms of Mean Squared Error (MSE), Convergence rate and accuracy.

Acknowledgments. This work is supported by University of Malaya High Impact Research Grant no vote UM.C/625/HIR/MOHE/SC/13/2 from Ministry of Higher Education Malaysia.

References

1. Radhika, Y., Shashi, M.: Atmospheric Temperature Prediction using Support Vector Machines. Int. J. of Computer Theory and Engineering 1(1), 1793–8201 (2009)
2. Akcayol, M.A., Cinar, C.: Artificial neural network based modeling of heated catalytic converter performance. J. Applied thermal Engineering 25, 2341–2350 (2005)
3. Shereef, K.I., Baboo, S.S.: A New Weather Forecasting Technique using Back Propagation Neural Network with Modified Levenberg-Marquardt Algorithm for Learning. Int. J. of Computer Science 8(6-2), 1694–1814 (2011)
4. Kosko, B.: Neural Network and Fuzzy Systm, 1st edn. Prentice Hall of India (1994)

5. Krasnopolsky, V.M., Chevallier, F.: Some Neural Network application in environmental sciences. Part II: Advancing Computational Efficiency of environmental numerical models. J. Neural Networks 16(3-4), 335–348 (2003)
6. Coppin, B.: Artificial Intelligence Illuminated, USA. Jones and Bartlet Illuminated Series, ch. 11, pp. 291–324 (2004)
7. Basheer, I.A., Hajmeer, M.: Artificial neural networks: fundamentals, computing, design, and application. J. of Microbiological Methods 43(1), 3–31 (2000)
8. Zheng, H., Meng, W., Gong, B.: Neural Network and its Application on Machine fault Diagnosis. In: ICSYSE, pp. 576–579 (1992)
9. Rehman, M.Z., Nawi, N.M.: Improving the Accuracy of Gradient Descent Back Propagation Algorithm (GDAM) on Classification Problems. Int. J. of New Computer Architectures and their Applications (IJNCAA) 1(4), 838–847 (2012)
10. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning Representations by Back–Propagating Errors. J. Nature 323, 533–536 (1986)
11. Lahmiri, S.: A comparative study of backpropagation algorithms in financial prediction. Int. J. of Computer Science, Engineering and Applications (IJCSEA) 1(4) (2011)
12. Nawi, M.N., Ransing, R.S., AbdulHamid, N.: BPGD-AG: A New Improvement of Back-Propagation Neural Network Learning Algorithms with Adaptive Gain. J. of Science and Technology 2(2) (2011)
13. Wam, A., Esm, S., Esa, A.: Modified Back Propagation Algorithm for Learning Artificial Neural Networks. In: 8th NRSC, pp. 345–352 (2001)
14. Wen, J., Zhao, J.L., Luo, S.W., Han, Z.: The Improvements of BP Neural Network Learning Algorithm. In: 5th WCCC-ICSP, vol. 3, pp. 1647–1649 (2000)
15. Lahmiri, S.: Wavelet transform, neural networks and the prediction of s & p price index: a comparative paper of back propagation numerical algorithms. J. Business Intelligence 5(2) (2012)
16. Nawi, N.M., Ransing, R.S., Salleh, M.N.M., Ghazali, R., Hamid, N.A.: An improved back propagation neural network algorithm on classification problems. In: Zhang, Y., Cuzzocrea, A., Ma, J., Chung, K.-i., Arslan, T., Song, X. (eds.) DTA and BSBT 2010. CCIS, vol. 118, pp. 177–188. Springer, Heidelberg (2010)
17. Gupta, J.N.D., Sexton, R.S.: Comparing back propagation with a genetic algorithm for neural network training. J. International Journal of Management Science 27, 679–684 (1999)
18. Rehman, M.Z., Nawi, N.M.: Studying the Effect of adaptive momentum in improving the accuracy of gradient descent back propagation algorithm on classification problems. J. Int. Journal of Modern Physics: Conference Series 9, 432–439 (2012)
19. Yam, J.Y.F., Chow, T.W.S.: Extended least squares based algorithm for training feed forward networks. J. IEEE Transactions on Neural Networks 8, 806–810 (1997)
20. Yam, J.Y.F., Chow, T.W.S.: A weight initialization method for improving training speed in feed forward neural networks. J. Neurocomputing 30, 219–232 (2000)
21. Yam, J.Y.F., Chow, T.W.S.: Feed forward networks training speed enhancement by optimal initialization of the synaptic coefficients. J. IEEE Transactions on Neural Networks 12, 430–434 (2001)
22. Kwok, T.Y., Yeung, D.Y.: Objective functions for training new hidden units in constructive neural networks. J. IEEE Transactions on Neural Networks 8, 1131–1147 (1997)
23. Zhang, J., Lok, T., Lyu, M.: A hybrid particle swarm optimization back propagation algorithm for feed forward neural network training. J. Applied Mathematics and Computation 185, 1026–1037 (2007)

24. Shah, H., Ghazali, R., Nawi, N.M., Deris, M.M.: Global hybrid ant bee colony algorithm for training artificial neural networks. In: Murgante, B., Gervasi, O., Misra, S., Nedjah, N., Rocha, A.M.A.C., Taniar, D., Apduhan, B.O. (eds.) ICCSA 2012, Part I. LNCS, vol. 7333, pp. 87–100. Springer, Heidelberg (2012)
25. Shah, H., Ghazali, R., Nawi, N.M.: Hybrid ant bee colony algorithm for volcano temperature prediction. In: Chowdhry, B.S., Shaikh, F.K., Hussain, D.M.A., Uqaili, M.A. (eds.) IMTIC 2012. CCIS, vol. 281, pp. 453–465. Springer, Heidelberg (2012)
26. Karaboga, D.: Artificial bee colony algorithm. J. Scholarpedia 5(3), 6915 (2010)
27. Yao, X.: Evolutionary artificial neural networks. J. International Journal of Neural Systems 4(3), 203–222 (1993)
28. Mendes, R., Cortez, P., Rocha, M., Neves, J.: Particle swarm for feedforward neural network training. In: Proceedings of the International Joint Conference on Neural Networks, vol. 2, pp. 1895–1899 (2002)
29. Lonen, I., Kamarainen, I.J., Lampinen, J.I.: Differential Evolution Training Algorithm for Feed-Forward Neural Networks. J. Neural Processing Letters 17(1), 93–105 (2003)
30. Liu, Y.-P., Wu, M.-G., Qian, J.-X.: Evolving Neural Networks Using the Hybrid of Ant Colony Optimization and BP Algorithms. In: Wang, J., Yi, Z., Żurada, J.M., Lu, B.-L., Yin, H. (eds.) ISNN 2006. LNCS, vol. 3971, pp. 714–722. Springer, Heidelberg (2006)
31. Khan, A.U., Bandopadhyaya, T.K., Sharma, S.: Comparisons of Stock Rates Prediction Accuracy using Different Technical Indicators with Back propagation Neural Network and Genetic Algorithm Based Back propagation Neural Network. In: The First International Conference on Emerging Trends in Engineering and Technology, pp. 575–580 (2008)
32. Yang, X.S., Deb, S.: Cuckoo search via Lévy flights. In: World Congress on Nature & Biologically Inspired Computing, India, pp. 210–214 (2009)
33. Yang, X.S., Deb, S.: Engineering Optimisation by Cuckoo Search. J. of Mathematical Modelling and Numerical Optimisation 1(4), 330–343 (2010)
34. Tuba, M., Subotic, M., Stanarevic, N.: Modified cuckoo search algorithm for unconstrainedoptimization problems. In: The European Computing Conference, pp. 263–268 (2011)
35. Tuba, M., Subotic, M., Stanarevic, N.: Performance of a Modified Cuckoo Search Algorithm for Unconstrained Optimization Problems. J. Faculty of Computer Science 11(2), 62–74 (2012)
36. Chaowanawate, K., Heednacram, A.: Implementation of Cuckoo Search in RBF Neural Network for Flood Forecasting. In: Fourth International Conference on Computational Intelligence, Communication Systems and Networks, pp. 22–26 (2012)
37. Pavlyukevich, I.: Levy flights, non-local search and simulated annealing. J. of Computational Physics 226(2), 1830–1844 (2007)
38. Walton, S., Hassan, O., Morgan, K., Brown, M.: Modified cuckoo search: A new gradient free optimisation algorithm. J. Chaos, Solitons& Fractals 44(9), 710–718 (2011)
39. Hagan, M.T., Menhaj, M.B.: Training Feedforward Networks with the Marquardt Algorithm. J. IEEE Transactions on Neural Networks 5(6), 989–993 (1994)
40. Nawi, N.M., Khan, A., Rehman, M.Z.: A New Cuckoo Search based Levenberg-Marquardt (CSLM) Algorithm. In: Murgante, B., Misra, S., Carlini, M., Torre, C.M., Nguyen, H.-Q., Taniar, D., Apduhan, B.O., Gervasi, O. (eds.) ICCSA 2013, Part I. LNCS, vol. 7971, pp. 438–451. Springer, Heidelberg (2013)
41. Nawi, N.M., Khan, A., Rehman, M.Z.: A New Back-propagation Neural Network optimized with Cuckoo Search Algorithm. In: Murgante, B., Misra, S., Carlini, M., Torre, C.M., Nguyen, H.-Q., Taniar, D., Apduhan, B.O., Gervasi, O. (eds.) ICCSA 2013, Part I. LNCS, vol. 7971, pp. 413–426. Springer, Heidelberg (2013)
42. Nawi, N.M., Khan, A., Rehman, M.Z.: A New Optimized Cuckoo Search Recurrent Neural Network (CSRNN) Algorithm. In: Sakim, H.A.M., Mustaffa, M.T. (eds.) ROVISP. LNEE, vol. 291, pp. 335–341. Springer, Heidelberg (2013)

CSLMEN: A New Cuckoo Search Levenberg Marquardt Elman Network for Data Classification

Nazri Mohd Nawi[1,2], Abdullah Khan[2], M.Z. Rehman[2],
Tutut Herawan[3,4], and Mustafa Mat Deris[2]

[1] Software and Multimedia Centre (SMC)
[2] Faculty of Computer Science and Information Technology
Universiti Tun Hussein Onn Malaysia
86400, Parit Raja, Batu Pahat, Johor, Malaysia
[3] Department of Information System
University of Malaya
50603 Pantai Valley, Kuala Lumpur, Malaysia
[4] AMCS Research Center, Yogyakarta, Indonesia
{nazri,mmustafa}@uthm.edu.my, hi100010@siswa.uthm.edu.my,
zrehman862060@gmail.com, tutut@um.edu.my

Abstract. Recurrent Neural Networks (RNN) have local feedback loops inside the network which allows them to store earlier accessible patterns. This network can be trained with gradient descent back propagation and optimization technique such as second-order methods. Levenberg-Marquardt has been used for networks training but still this algorithm is not definite to find the global minima of the error function. Nature inspired meta-heuristic algorithms provide derivative-free solution to optimize complex problems. This paper proposed a new meta-heuristic search algorithm, called Cuckoo Search (CS) to train Levenberg Marquardt Elman Network (LMEN) in achieving fast convergence rate and to avoid local minima problem. The proposed Cuckoo Search Levenberg Marquardt Elman Network (CSLMEN) results are compared with Artificial Bee Colony using BP algorithm, and other hybrid variants. Specifically 7-bit parity and Iris classification datasets are used. The simulation results show that the computational efficiency of the proposed CSLMEN training process is highly enhanced when coupled with the Cuckoo Search method.

Keywords: Cuckoo Search, Levenberg Marquardt algorithm, Neural network, Data Classification.

1 Introduction

Artificial Neural Network (ANN) is a unified group of artificial neurons that uses a mathematical or computational form for information processing based on a connectionist network calculation. In most cases, ANN is an adaptive construction that changes its formation based on outer or inner information that flows in the network. They can be used to copy composite affairs between inputs and outputs or to

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_17, © Springer International Publishing Switzerland 2014

find patterns in data [1]. Among some possible network architectures the ones usually used are the feed forward and the recurrent neural networks (RNN). In a feed forward neural network the signals propagate only in one direction, starting from the input layer, through the hidden layers to the output layer. While RNN have local feed-back loops inside the network which allows them to store earlier accessible patterns. This ability makes RNN more advanced than the conventional feed forward neural networks in modeling active systems because the network outputs are functions of both the present inputs as well as their inner states [2-3].

Earlier, many types of RNN have been proposed and they are either moderately recurrent or fully recurrent. RNN can carry out non-linear bouncy mappings and have been used in many applications including associative memories, spatiotemporal pattern classification, managed optimization, forecasting and simplification of pattern sequences [4-8]. Fully recurrent networks use unrestricted fully interrelated architectures and learning algorithms that can deal with time altering input and output in non-minor manners. However, certain property of RNN makes many of algorithms less efficient, and it often takes an enormous amount of time to train a network of even a reasonable size. In addition, the complex error surface of the RNN network makes many training algorithms more flat to being intent in local minima. Thus the main disadvantage of the RNN is that they require substantially more connections, and more memory in simulation, then standard back propagation network, thus resulting in a substantial increase in computational times. Therefore, this study used partial recurrent networks, whose connections are mainly feed forward, but they comprise of carefully selected set of feedback associates. The reappearance allows the system to memorize past history from the precedent without complicating the learning extremely [9]. One example of such a recurrent network is an Elman which is set up as a usual feed forward network [10].

The Elman network can be cultured with gradient descent back propagation and optimization technique, like normal feed forward neural networks. The back propagation has several problems for numerous applications. The algorithm is not definite to find the global minimum of the error function since gradient descent may get stuck in local minima, where it may stay indeterminately [11-13, 24]. In recent years, a number of research studies have attempted to surmount these problems and to improve the convergence of the back propagation. Optimization methods such as second-order conjugate gradient, quasi-Newton, and Levenberg-Marquardt have also been used for networks training [14, 15]. To overcome these drawbacks many evolutionary computing techniques have been used. Evolutionary computation is often used to train the weights and parameters of the networks. One such example of evolutionary algorithms is Cuckoo search (CS) developed by Yang and Deb [16] used for global optimization [17-19]. The CS algorithm has been applied independently to solve several engineering design optimization problems, such as the design of springs and welded beam structures [20], and forecasting [21].

In this paper, we proposed a new meta-hybrid search algorithm, called Cuckoo Search Levenberg Marquardt Elman Network (CSLMEN). The convergence performance of the proposed CSLMEN algorithm is analyzed on XOR and OR datasets in terms of simulations. The results are compared with artificial bee colony using back-propagation (ABCBP) algorithm and similar hybrid variants. The main goal is to decrease the computational cost and to accelerate the learning process in RNN using Cuckoo Search and Levenberg-Marquardt algorithms.

The remaining paper is organized as follows: Section 2 gives literature review on Levenberg Marquardt algorithm, explains Cuckoo Search via levy flight and CSLMEN algorithm is proposed. The simulation results are discussed in Section 3 and finally the paper is concluded in the Section 4.

2 Training Algorithms

2.1 Levenberg Marquardt (LM) Algorithm

To Speed up the convergence, Levenberg-Marquardt algorithm is used. The Levenberg-Marquardt (LM) algorithm is an approximation to Newton's method to accelerate training speed. Benefits of applying LM algorithm over variable learning rate and conjugate gradient method were reported in [22]. The LM algorithm is developed through Newton's method.

For Gauss-Newton Method:

$$\nabla w = -[J^T(t)J(t)]^{-1}J(t)e(t). \tag{1}$$

For the Levenberg-Marquardt algorithm as the variation of Gauss-Newton Method:

$$\mathrm{w(k+1) = w(k)} - [J^T(t)J(t) + \mu I]^{-1}J(t)e(t), \tag{2}$$

where $\mu > 0$ and is a constant; I is identity matrix. So that the algorithm will approach Gauss- Newton, which should provides faster convergence.

2.2 Cuckoo Search (CS) Algorithm

Cuckoo Search (CS) algorithm is a meta-heuristic technique proposed by Xin-She Yang [16, 17]. This algorithm was inspired by the obligate brood parasitism of some cuckoo species in which they lay their eggs in the other bird's nest. Some host birds don't recognize any difference in the eggs of the cuckoo and its own. But, if the host bird finds an eggs as not their own, they will simply throw the unknown eggs or simply leave its nest to build a new nest elsewhere. Some other cuckoo species have evolved in such a way that female parasitic cuckoos are very specialized in mimicking the color and pattern of the eggs of the chosen host species. This reduces the probability of their eggs being abandoned and thus increases their survival rate. The CS algorithm follows the three idealized rules:

a) Each cuckoo lays one egg at a time, and put its egg in randomly chosen nest;
b) The best nests with high quality of eggs will carry over to the next generations;
c) The number of available host nests is fixed, and the egg laid by a cuckoo is discovered by the host bird with a probability $pa \in [0, 1]$.

3 The Proposed CSLMEN Algorithm

The proposed Cuckoo Search Levenberg Marquardt Elman Network (CSLMEN) is implemented in two stages. In the first phase the Cuckoo Search algorithm via levy flight is initialized with nests and then CS is integrated into Levenberg Marquardt Elman network. In the second phase, the best nests as weights are used for the Levenberg Marquardt Elman network training. In this proposed model the CS is responsible to modify the perimeters for the convergences of this algorithm. Then the overall procedure automatically adjusts its own parameters and converge to global minimum. In the proposed CSLMEN algorithm, the CS algorithm is initiated with a set of randomly generated nests as weights for the networks. Then each randomly generated weight is passed to Levenberg Marquardt back propagation Elman network for further process. The sum of square error is considered as a performances index for the proposed CSLMEN algorithm. So the weight value of a matrix is calculated as following;

$$W_t = \sum_{c=1}^{m} a.(rand - \tfrac{1}{2}). \tag{3}$$

$$B_t = \sum_{c=1}^{m} a.(rand - \tfrac{1}{2}) , \tag{4}$$

where, $W_t = t^{th}$ is the weight value in a weight matrix. the rand in the Equation (3) is the random number between[0 1]. a is constant parameter and for the proposed method its value is less than one. And B_t bias value. So the list of weight matrix is as follows.

$$W^s = [W_t^{\ 1}, W_t^{\ 2}, W_t^{\ 3}, \ldots, W_t^{\ n-1}] . \tag{5}$$

Now from back propagation process sum of square error can be easily calculated for every weight matrix in W^s. So the error can be calculated as:

$$e_r = (T_r - X_r). \tag{6}$$

The performances index for the network is calculated as;

$$V(x) = \tfrac{1}{2} \sum_{r=1}^{R} (T_r - X_r)^T (T_r - X_r). \tag{7}$$

In the proposed CSLMEN, the average sum of square is considered as the performance index and calculated as following;

$$V_\mu(x) = \frac{\sum_{j=1}^{N} V_F(x)}{P_i} , \tag{8}$$

where, $V_\mu(x)$ is the average performance, $V_F(x)$ is the performance index, and P_i is the number of cuckoo population in i^{th} iteration. The weight and bias are calculated

according to the back propagation method. The sensitivity of one layer is calculated from its previous one and the calculation of the sensitivity start from the last layer of the network and move backward. To speed up convergence Levenberg Marquardt is selected a learning algorithm. The Levenberg-Marquardt (LM) algorithm is an approximation to Newton's method to get faster training speed.
Assume the error function is:

$$E(t) = \frac{1}{2}\sum_{i=1}^{N} e_r{}^2(t), \tag{9}$$

where, $\mathrm{e(t)}$ is the error; N is the number of vector elements, and $\mathrm{E(t)}$ is the sum of spares function then the gradient is calculated as:

$$\nabla E(t) = J^T(t)e(t), \tag{10}$$

$$\nabla^2 E(t) = J^T(t)J(t), \tag{11}$$

Where, $\nabla \mathrm{E(t)}$ is the gradient; $\nabla^2 \mathrm{E(t)}$ is the Hessian matrix of E (t), and J (t) is the Jacobin matrix which is calculated in Equation (12);

$$\mathrm{J\,(t)} = \begin{bmatrix} \frac{\partial \mathrm{e_1(t)}}{\partial \mathrm{t_1}} & \frac{\partial \mathrm{e_1(t)}}{\partial \mathrm{t_2}} \cdots\cdots & \frac{\partial \mathrm{e_1(t)}}{\partial \mathrm{t_n}} \\ \frac{\partial \mathrm{e_2(t)}}{\partial \mathrm{t_1}} & \frac{\partial \mathrm{e_2(t)}}{\partial \mathrm{t_2}} \cdots\cdots & \frac{\partial \mathrm{e_2(t)}}{\partial \mathrm{t_n}} \\ & \cdot & \\ & \cdot & \\ & \cdot & \\ \frac{\partial \mathrm{e_n(t)}}{\partial \mathrm{t_1}} & \frac{\partial \mathrm{e_n(t)}}{\partial \mathrm{t_2}} \cdots\cdots & \frac{\partial \mathrm{e_n(t)}}{\partial \mathrm{t_n}} \end{bmatrix} \tag{12}$$

For Gauss-Newton Method, the following Equation is used:

$$\nabla w = -[J^T(t)J(t)]^{-1}J(t)e(t). \tag{13}$$

For the Levenberg-Marquardt algorithm as the variation of Gauss-Newton Method:

$$w(k+1) = w(k) - [J^T(t)J(t) + \mu I]^{-1}J(t)e(t), \tag{14}$$

where $\mu > 0$ and is a constant; I is identity matrix. So that the algorithm will approach Gauss- Newton, which should provide faster convergence. Note that when parameter λ is large, the above expression approximates gradient descent (with a learning rate 1/λ) while for a small λ, the algorithm approximates the Gauss- Newton method.

The Levenberg-Marquardt works in the following manner:

a) It Presents all inputs to the network and compute the corresponding network outputs and errors using Equation (6 and 7) over all inputs. And compute sum of square of error over all input.

b) Compute the Jacobin matrix using Equation (12).
c) Solve Equation (13) to obtain ∇w .
d) Recomputed the sum of squares of errors using Equation (14) if this new sum of squares is smaller than that computed in step 1, then reduce μ by λ, update weight using $w(k+1) = w(k) - \nabla w$ and go back to step 1. If the sum of squares is not reduced, then increase μ by λ and go back to step 3.
e) The algorithm is assumed to have converged when the norm of the gradient Equation (10) is less than some prearranged value, or when the sum of squares has been compact to some error goal.

At the end of each epoch the list of average sum of square error of i^{th} iteration SSE can be calculated as;

$$SSE_i = \{V_\mu{}^1(x), V_\mu{}^2(x), V_\mu{}^3(x) \ldots\ldots V_\mu{}^n(x)\} \tag{15}$$

$$x_j = Min\{V_\mu{}^1(x), V_\mu{}^2(x), V_\mu{}^3(x) \ldots\ldots V_\mu{}^n(x)\}, \tag{16}$$

and the rest of the average sum of square is consider as other Cuckoo nest. A new solution x_i^{t+1} for Cuckoo i is generated using a levy flight in the following Equation;

$$x_i^{t+1} = x_i^t + \alpha \otimes \text{levy}(\lambda). \tag{17}$$

So the movement of the other Cuckoo x_i toward x_j can be drawn from Equation (18);

$$X = \begin{cases} x_i + \text{rand} \cdot (x_j - x_i) & \text{rand}_i > p_\alpha \\ x_i & \text{else} \end{cases}. \tag{18}$$

The Cuckoo Search can move from x_i toward x_j through levy fight;

$$\nabla X_i = \begin{cases} x_i + \alpha \otimes levy(\lambda)\,(x_j - x_i) & rand_i > p_\alpha \\ x_i & else \end{cases}, \tag{19}$$

where ∇V_i is a smal movement of x_i toward x_j. The weights and biases for each layer can then be adjusted as;

$$W_n{}^{t+1} = U_n{}^{t+1} = W_n{}^t - \nabla X_i \tag{20}$$

$$B_n{}^{t+1} = B_n{}^t - \nabla X_i \tag{21}$$

4 Results and Discussions

Basically, the main focus of this paper is to compare the performance of different algorithms in reducing the error and accuracy in network convergence. Some simulation results, tools and technologies, network topologies, testing methodology and the classification problems used for the entire experimentation will be discussed further in the this section.

4.1 Preliminary Study

Iris and 7-Parity bit benchmark classification problem were used for testing the performance of the proposed CSLMEN with other algorithms. The simulation experiments were performed on a 1.66 GHz AMD Processor with 2GB of RAM and by using Matlab 2009b Software. For each algorithm, 20 trials were run during the simulation process. In this paper, the following algorithms are compared, analyzed and simulated on Iris and 7-Parity bit benchmark classification problem

a) Artificial Bee Colony Algorithm based Back Propagation (ABCBP) algorithm,
b) Artificial Bee Colony Algorithm based Levenberg Marquardt (ABCLM) algorithm,
c) Back Propagation Neural Network (BPNN)algorithm, and
d) The Proposed Cuckoo Search based Elman Network (CSLMEN) algorithm.

The performance measure for each algorithm is based on the Mean Square Error (MSE), standard deviation (SD) and accuracy. The three layers feed forward neural network architecture (i.e. input layer, one hidden layer, and output layers.) is used for each problem. The number of hidden nodes is keep fixed to 5. In the network structure, the bias nodes are also used and the log-sigmoid activation function is applied. While the target error and the maximum epochs are set to 0.00001 and 1000. The learning value is set to 0.4. A total of 20 trials are run for each dataset. The network results are stored in the result file for each trial. CPU time, average accuracy, and Mean Square Error (MSE) are recorded for each independent trial on Iris and 7-Parity bit benchmark classification problem.

4.2 Iris Benchmark Classification Problems

The Iris classification dataset is the most famous dataset to be found in the pattern recognition literature. It consists of 150 instances, 4 inputs, and 3 outputs in this dataset. The classification of Iris dataset involves the data of petal width, petal length, sepal length, and sepal width into three classes of species, which consist of Iris Santos, Iris Versicolor, and Iris Virginia.

Table 1. CPU Time, Epochs, MSE, Accuracy, Standard Deviation (SD) for 4-5-3 ANN Architecture

Algorithm	**ABCBP**	**ABCLM**	**BPNN**	**CSLMEN**
CPU Time	156.43	171.52	168.47	7.07
Epochs	1000	1000	1000	64
MSE	0.155	0.058	0.311	3.2E-06
SD	0.022	0.005	0.022	2.2E-06
Accuracy (%)	86.87	79.55	87.19	99.9

Table 1 shows that the proposed CSLMEN method performs well on Iris dataset. The CSLMEN converges to global minima in 7.07 CPU seconds with an average accuracy of 99.9 percent and achieves a MSE and SD of 3.2E-06, 2.2E-06 respectively. While other algorithms fall behind in-terms of MSE, CPU time and accuracy. The proposed CSLMEN performs well and converges to global minima within 64 epochs for the 4-5-3 network structure as compared to other algorithms which take more CPU time and epochs to converge.

4.3 Seven Bit-Parity Problems

The parity problem is one of the most popular initial testing task and a very demanding classification dataset for neural network to solve. In parity problem, if a give input vectors contains an odd number of one, the corresponding target value is 1, and otherwise the target value is 0. The N-bit parity training set consist of 2N training pairs, with each training pairs comprising of an N-length input vector and a single binary target value. The 2N input vector represents all possible combinations of the N binary numbers. Here the Neural Network Architecture used is 7-5-1.

Table 2. CPU Time, Epochs, MSE, Accuracy, Standard Deviation (SD) for 7-5-1 ANN Architecture

Algorithm	**ABCBP**	**ABCLM**	**BPNN**	**CSLMEN**
CPU Time	183.39	134.88	142.07	0.60
Epochs	1000	1000	1000	6
MSE	0.12	0.08	0.26	2.2E-06
SD	0.008	0.012	0.014	2.8E-06
Accuracy (%)	82.12	69.13	85.12	99.98

From Table 2, it can be seen that that there is eventually a decrease in the CPU time, number of epochs, the mean square error using CSLMEN algorithm. While the accuracy for the 7-parity bit classification problem with five hidden neurons is also increased considerably using the CSLMEN algorithm. The proposed CSLMEN converged to a MSE of 2.2E-06 within 6 epochs. The ABCLM algorithm has an MSE of 0.08 and the ABCBP has the MSE of 0.12 with 82.12 and 69.13 % of accuracy. The simple BPNN still needs more epochs and CPU time to converge to global minima.

5 Conclusions

Elman Recurrent Neural Network (ERNN) is one of the most widely used and a popular training feed forward neural network. The Elman network can be cultured with gradient descent back propagation and optimization technique. In this paper, a new meta-heuristic search algorithm, called cuckoo search (CS) is proposed to train Levenberg Marquardt Elman Network (LMEN) to achieve fast convergence rate and to minimize the training error. The performance of the proposed CSLMEN algorithm is compared with the ABCLM, ABCBP and BPNN algorithms. The proposed CSLMEN model is verified by means of simulation on 7-bit parity and Iris classification datasets. The simulation results show that the proposed CSLMEN is far better than the previous methods in terms of MSE, convergence rate and accuracy.

Acknowledgments. This work is supported by University of Malaya High Impact Research Grant no vote UM.C/625/HIR/MOHE/SC/13/2 from Ministry of Higher Education Malaysia.

References

1. Firdaus, A.A.M., Mariyam, S.H.S., Razana, A.: Enhancement of Particle Swarm Optimization in Elman Recurrent Network with bounded Vmax Function. In: Third Asia International Conference on Modelling & Simulation (2009)
2. Peng, X.G., Venayagamoorthy, K., Corzin, K.A.: Combined Training of Recurrent Neural Networks with Particle Swarm Optimization and Back propagation Algorithms for Impedance Identification. In: IEEE Swarm Intelligence Symposium, pp. 9–15 (2007)
3. Haykin, S.: Neural Networks:A Comprehensive Foundation, 2nd edn., pp. 84–89 (1999) ISBN 0-13-273350
4. Ubeyli, E.D.: Recurrent neural networks employing lyapunov exponents for analysis of Doppler ultrasound signals. J. Expert Systems with Applications 34(4), 2538–2544 (2008)
5. Ubeyli, E.D.: Recurrent neural networks with composite features for detection of electrocardiographic changes in partial epileptic patients. J. Computers in Biology and Medicine 38(3), 401–410 (2008)
6. Saad, E.W., Prokhorov, D.V., WunschII, D.C.: Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks. J. IEEE Transactions on Neural Networks 9(6), 1456–1470 (1998)
7. Gupta, L., McAvoy, M.: Investigating the prediction capabilities of the simple recurrent neural network on real temporal sequences. J. Pattern Recognition 33(12), 2075–2081 (2000)
8. Gupta, L., McAvoy, M., Phegley, J.: Classification of temporal sequences via prediction using the simple recurrent neural network. J. Pattern Recognition 33(10), 1759–1770 (2000)
9. Nihal, G.F., Elif, U.D., Inan, G.: Recurrent neural networks employing lyapunov exponents for EEG signals classification. J. Expert Systems with Applications 29, 506–514 (2005)
10. Elman, J.L.: Finding structure in time. J. Cognitive Science 14(2), 179–211 (1990)
11. Rehman, M.Z., Nawi, N.M.: Improving the Accuracy of Gradient Descent Back Propagation Algorithm(GDAM) on Classification Problems. Int. J. of New Computer Architectures and their Applications (IJNCAA) 1(4), 838–847 (2012)
12. Wam, A., Esm, S., Esa, A.: Modified Back Propagation Algorithm for Learning Artificial Neural Networks. In: The 18th National Radio Science Conference, pp. 345–352 (2001)
13. Wen, J., Zhao, J.L., Luo, S.W., Han, Z.: The Improvements of BP Neural Network Learning Algorithm. In: 5th Int. Conf. on Signal Processing WCCC-ICSP, pp. 1647–1649 (2000)

14. Tanoto, Y., Ongsakul, W., Charles, O., Marpaung, P.: Levenberg-Marquardt Recurrent Networks for Long-Term Electricity Peak Load Forecasting. J. Telkomnika 9(2), 257–266 (2011)
15. Peng, C., Magoulas, G.D.: NonmonotoneLevenberg–Marquardt training of recurrent neural architectures for processing symbolic sequences. J. of Neural Comput& Application 20, 897–908 (2011)
16. Yang, X.S., Deb, S.: Cuckoo search via Lévy flights. In: Proceedings of World Congress on Nature & Biologically Inspired Computing, India, pp. 210–214 (2009)
17. Yang, X.S., Deb, S.: Engineering Optimisation by Cuckoo Search. J. International Journal of Mathematical Modelling and Numerical Optimisation 1(4), 330–343 (2010)
18. Tuba, M., Subotic, M., Stanarevic, N.: Modified cuckoo search algorithm for unconstrained optimization problems. In: Proceedings of the European Computing Conference (ECC 2011), pp. 263–268 (2011)
19. Tuba, M., Subotic, M., Stanarevic, N.: Performance of a Modified Cuckoo Search Algorithm for Unconstrained Optimization Problems. J. Faculty of Computer Science 11(2), 62–74 (2012)
20. Yang, X.S., Deb, S.: Engineering optimisation by cuckoo search. J. Int. J. Mathematical Modelling and Numerical Optimisation 1(4), 330–343 (2010)
21. Chaowanawate, K., Heednacram, A.: Implementation of Cuckoo Search in RBF Neural Network for Flood Forecasting. In: 4th International Conference on Computational Intelligence, Communication Systems and Networks, pp. 22–26 (2012)
22. Hagan, M.T., Menhaj, M.B.: Training Feedforward Networks with the Marquardt Algorithm. J. IEEE Transactions on Neural Networks 5(6), 989–993 (1999)
23. Nourani, E., Rahmani, A.M., Navin, A.H.: Forecasting Stock Prices using a hybrid Artificial Bee Colony based Neural Network. In: ICIMTR 2012, Malacca, Malaysia, pp. 21–22 (2012)
24. Rehman, M.Z., Nawi, N.M.: Studying the Effect of adaptive momentum in improving the accuracy of gradient descent back propagation algorithm on classification problems. J. Int. Journal of Modern Physics: Conference Series 9, 432–439 (2012)

Enhanced MWO Training Algorithm to Improve Classification Accuracy of Artificial Neural Networks

Ahmed A. Abusnaina[1], Rosni Abdullah[1], and Ali Kattan[2]

[1] School of Computer Sciences, Universiti Sains Malaysia, Penang, Malaysia
[2] IT Department, Ishik University, Erbil, Iraq
abusnaina@ymail.com, rosni@cs.usm.my,
ali.kattan@ishik.edu.iq

Abstract. The Mussels Wandering Optimization (MWO) algorithm is a novel meta-heuristic optimization algorithm inspired ecologically by mussels' movement behavior. The MWO algorithm has been used to solve linear and nonlinear functions and it has been adapted in supervised training of Artificial Neural Networks (ANN). Based on the latter application, the classification accuracy of ANN based on MWO training was on par with other algorithms. This paper proposes an enhanced version of MWO algorithm; namely Enhanced-MWO (E-MWO) in order to achieve an improved classification accuracy of ANN. In addition, this paper discusses and analyses the MWO and the effect of MWO parameters selection (especially, the shape parameter) on ANN classification accuracy. The E-MWO algorithm is adapted in training ANN and tested using well-known benchmarking problems and compared against other algorithms. The obtained results indicate that the E-MWO algorithm is a competitive alternative to other evolutionary and gradient-descent based training algorithms in terms of classification accuracy and training time.

Keywords: Mussels Wandering Optimization, Artificial Neural Networks, Supervised Training, Optimization, Meta-heuristic, Pattern Classification.

1 Introduction

Artificial Neural Network (ANN) is a simplified mathematical approximation of biological neural networks in terms of structure and function [1]. ANN has the ability to do as human-like brain tasks, such as identification of objects and patterns, making decisions based on prior experiences and knowledge, prediction of future events based on past experience [2]. The mechanism of training algorithms deals with adjusting the ANN weights' values in order to minimize the error function and the mechanism of information flow that depends on ANN structure are the most challenging aspects of ANN [1], [4], [5].

Pattern classification is one of many applications that are addressed by neural networks [6], [7], [8]. Pattern classification is concerned with assigning an object to a predefined class based on a number of feature attributes related to that object.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_18, © Springer International Publishing Switzerland 2014

It shows how machines can observe the environment and learn to distinguish patterns and make reasonable decisions about the patterns categories [9], [10].

The training process of ANN deals with adjusting and altering the weights and/or structure of the network depending on a specific training algorithm. The ANN weights search space is considered as continuous optimization problem because it is high-dimensional and multimodal, also it could be corrupted by noises or missing data [15], [17]. The training is process of searching for suitable values in this search space.

A number of metaheuristic and global optimization algorithms have been proposed during the last twenty years. One recent ecologically inspired meta-heuristic is the Mussels Wandering Optimization algorithm (MWO). The MWO algorithm was developed by An et. al. [3] inspired by mussels' movement behavior when they form bed pattern in their surroundings habitat. Stochastic decision and Le´vy walk are two evolutionary mechanisms used mathematically to formulate a landscape-level of mussels' pattern distribution. However, the literature lacks the deep analysis of how these mechanisms affect the MWO performance.

Several linear and nonlinear mathematical functions have been solved by the MWO proposed by An et. al. [3]. In addition, the MWO algorithm has been successfully adapted in supervised training of ANN for pattern classification of several benchmarking problems [5]. In [5] the authors argue that the MWO outperforms other algorithms in terms of ANN training time. The classification accuracy however was on par. The proposed method focuses on improving the classification accuracy of ANN for pattern classification. The E-MWO adaptively sets the algorithm parameters, uses a new selection scheme, depends on multi-step length for updating the mussels' positions and it terminates depending on the dynamic quality measure of mussels candidates solutions.

In this paper, the enhanced version of MWO; the E-MWO algorithm is proposed and demonstrated. The E-MWO algorithm is supposed to overcome the weakness of the MWO algorithm by obtaining better performance results in terms of classification accuracy and training time. This paper tries to answer the following research questions: How do the parameters of MWO affects the classification accuracy of ANN? How can the MWO algorithm be enhanced to increase its performance?

The rest of this paper is organized as follows: section 2 presents some related works and introduces the MWO algorithm. Section 3 demonstrates the E-MWO algorithm and its adaptation for training the ANN, while section 4 presents the experimental results. Section 5 concludes the paper.

2 Related Works

The supervised training model of ANN is able to map a set of inputs to a set of outputs, eventually the ANN will be able to predict the output class of inputs when their desired output is unknown [11]. Several algorithms have been proposed for supervised training of ANN. Gradient-descent technique such as Back-propagation (BP) is considered the most well-known algorithm [26]. BP suffers from its slowness [12], [13] and the possibility of falling into local minima [14]. The drawbacks of

Gradient-descent algorithms have been handled by meta-heuristic algorithms that depend on global optimization methods [14], [15]. Such training algorithms are: Genetic algorithm (GA) [16], [27] Artificial Bee Colony (ABC) [15], Group Search Optimizer (GSO) [17], [18], Particle Swarm Optimization (PSO) [19] and the Harmony Search (HS) algorithm [14], [20], [21].

The MWO algorithm is initialized with a population of random candidate solutions. Each mussel is assigned a randomized position then it iteratively moves through the problem space. The mussels' population is attracted towards the location of the best mussel (the best mussel is the mussel that has the highest fitness value) achieved so far across the whole population.

The MWO algorithm is composed of six steps as follows: (1) Initialization of mussels' population and the algorithm parameters. (2) Calculation of the short-range density ζ_s and the long-range density ζ_l for each mussel. (3) Determination of the movement strategy for each mussel. (4) Update process of the position for all mussels. (5) Evaluation of the fitness for each mussel after position updating. (6) Examination of the termination criteria. The MWO algorithm is shown in Algorithm1, where the list of equations used by the MWO is presented in APPENDIX A.

Algorithm 1. MWO algorithm

```
Initialization:
Set $t = 0$;
FOR (mussel $m_i$, $i = 1$ to $N$)
    Uniformly randomize initial position $x_i(0)$for all mussel $m_i$;
    Calculate the fitness value of initial mussels : $f(x_i(0))$;
END FOR
Find the global best mussel and record its position as $x_g$;
Iteration: // G: maximum number of iterations, $\epsilon$: predefined precision.
WHILE ($t<G$, or $f(x^*) >\epsilon$)
    FOR (mussel $m_i$, $i = 1$ to $N$)
        Calculate the distance from $m_i$ to other mussels by Eq. (1);
        Calculate the short-range reference and long-range reference by Eq.(2);
        Calculate short-range density and long-range density by Eq.(3) and Eq.(4);
        Compute its moving probability $P_i(t)$ according to Eq.(5);
        IF $P_i = 1$ THEN Generate its step length $\ell_i$ (t) Le´vy distribution by Eq.(6)
        ELSE $\ell_i (t) = 0$
        END IF
         Compute the new position coordinate $\grave{x}_i (t)$ using Eq. (7).
         Evaluate the fitness value of the new position coordinate $f(\grave{x}_i (t))$
    END FOR
    Rank all mussels according to the fitness from best to worst, find the global best
        mussel and update the best position $x_{g;}$
    Set $t = t+1$;
END WHILE
Output the optimized results and end the algorithm
```

Through the MWO algorithm, the population of mussels consists of N individuals, these individuals are in a certain spatial region of marine ''bed'' called the habitat. The habitat is mapped to a d-dimensional space S^d of the problem to be optimized, where the objective function value $f(s)$ at each point $s \in S^d$ represents the nutrition provided by the habitat. Each mussel has a position $x_i:=(x_{i1}, . . ., x_{id})$; $i \in \mathrm{N}_N = \{1,2,3, . . ., N\}$ in S^d, which therefore, they form a specified spatial bed pattern.

3 The Proposed Method

This section explains the enhanced version of the MWO algorithm, namely the E-MWO algorithm, by introducing the new added or modified parts over the MWO only. The MWO is considered as competitive alternative to other existing algorithms for training the ANN in terms of training time [5]. However, after performing many empirical experiments the MWO terminated prematurely due to high selective pressure on global best mussel (Lines: 7, 18 & 21 in Algorithm 1). Thus, the E-MWO tries to solve this problem by introducing a new hybrid-selection scheme (Lines: 7, 19, 25, 26 & 27 in Algorithm 2). The MWO depends on single step to update the mussel position (Line 15 in Algorithm 1). In order to make the algorithm more explorative to the solution space, the E-MWO depends on multi-step length (Line 16 in Algorithm 2). The set of parameters in the MWO must be determined statically before run time. However, some of these parameters highly affect the solution quality, especially the shape parameter (μ) [3][5]. The E-MWO set the value of μ dynamically and adaptively depending on the dynamic quality of the solutions (Lines: 22, 23 & 24 in Algorithm 2). The MWO depends mainly on the number of iterations or predefined precision as termination condition (Line 9 in Algorithm 1). Thus, even if the solution is not good enough the MWO will continue its run until it reaches the whole number of iterations, this behavior waste the computational time. The E-MWO has the property to terminate its run depending on the dynamic solution quality instead of the number of iterations only (Line 9 in Algorithm 2). The E-MWO algorithm is shown in Algorithm 2, where the equations from Eq. 1 to Eq.7 used by the E-MWO are same as in the MWO which presented in APPENDIX A. The complexity of E-MWO is O(t*N), where t is the last iteration number that the algorithm reached before termination condition satisfied and N is the population size. The subsequent subsections explain all the E-MWO features in details.

3.1 New Selection Scheme to Guide the Population

The original MWO algorithm uses the global best mussel as guidance to update all other mussels (see Eq. 7 in APPENDIX A). However, this selection scheme is good at the beginning of the optimization process. But based on empirical experiments, the global best mussel becomes stable after some iterations (i.e. the same global best mussel with same fitness value do not change), so the convergence could not be improved further.

This problem is solved in E-MWO by depending on two selection schemes simultaneously: global best and random selection schemes. At each iteration, the mussels' population will update their ANN weights values as long as there is a new mussel becomes global best or the global best has new fitness value. But if the global best is repeated for *t* iterations with the same fitness value, another mussel is chosen randomly from the best *M* mussels as a guidance mussel. The randomly selected mussel will be used in the update process, thus new solution regions can be explored.

Algorithm 2. E-MWO algorithm

```
Initialization:
Set $t = 0$;
FOR (mussel $m_i$, $i = 1$ to $N$)
    Uniformly randomize the initial position $x_i(0)$for all mussel $m_i$
    Calculate the initial fitness value of  the mussel $f(x_i(0))$
END FOR
Find the global best mussel and record its position as $x_g$ and set it as selected mussel $m_s$
Iteration:
WHILE (t <MaxIterations AND $U_R > \varepsilon_1$ AND $S_R < \varepsilon_2$)
    FOR (mussel $m_i$, $i = 1$ to $N$)
        Calculate the distance from $m_i$ to all other mussels by Eq. (1);
        Calculate short-range reference and long-range reference by Eq.(2);
        Calculate short-range density and long-range density by Eq.(3) and Eq.(4);
        Compute the moving probability $P_i(t)$ according to Eq.(5);
        IF $P_i = 1$ THEN
            Generate all steps length $\ell_{ij}(t)$ Le´vy distribution by Eq.(6)
        ELSE  $\ell_{ij}(t) = 0$
        END IF
        Compute the new position coordinate $\grave{x}_i(t)$ using Eq. (7) according to $m_s$.
        Evaluate the fitness value of the new position coordinate $f(\grave{x}_i(t))$
    END FOR
    Calculate the Similarity Ratio $S_R$ by Eq.(8)
    Calculate the Update Ratio $U_R$ by Eq.(9)
    Calculate the new value of shape parameter ($\mu$) by Eq.(10)
    Rank all mussels by their fitness, find the global best mussel and set it as selected
        mussel $m_s$.
    IF ( the global best mussel is the same for the last $t$ iterations AND with same
        fitness value)
    THEN (select a mussel randomly from the best $M$ of the mussels' population and
        set it as selected mussel $m_s$ )
    Set $t = t+1$;
END WHILE
Output the optimized results and end the algorithm
```

3.2 Adaptive Setting of E-MWO Parameters

The shape parameter (μ) used in MWO to determine the movement strategy, which highly affects the mussel step-length to move from one position to another towards the best global mussel (see Eq. 6 in APPENDIX A). By using the original MWO in any optimization problem, many empirical experiments must be conducted to choose the best value of μ that can produce the best solution; i.e. high classification accuracy.

When μ is small; i.e. $1 < \mu \leq 1.5$, the step length is large and amorphous. Thus, the difference between the new updated position and the previous position of the mussel is large. This will make the mussels population in dispersion state and face difficulties to find the optimum solution. Thus, the classification accuracy is low when μ is small.

On the other hand, when μ is large, i.e. $1.9 \leq \mu < 3$, the step length is very small and its effect could be neglected. Thus, the difference between the new updated position and the previous position of the mussel is very small. This will make the mussels population nearly in the same initial position and face difficulties to find the optimum solution. Consequently, the classification accuracy is low when μ is large.

The adaptive feature of E-MWO will allow setting the value of μ dynamically and adaptively depending on new introduced variables which are: Similarity Ratio (S_R) and Update Ratio (U_R) as demonstrated in equations (8), (9) and (10).

$$S_R = \frac{\text{Number of Mussels have the same fitness value at iteration } t}{\text{Number of Mussels}} \tag{8}$$

$$U_R = \frac{\text{Number of Mussels update their NN weight values at iteration } t}{\text{Number of Mussels}} \tag{9}$$

$$\text{Shape Parameter } (\mu) = \mu_c + \lambda_1 S_R + \lambda_2 U_R \tag{10}$$

Where: μ_c is shape parameter constant, λ_1 and λ_2 coefficients.

After running many empirical experiments, when μ is very small or very high (i.e. $\mu<1.5$ and $\mu> 1.9$) the update ratio becomes very small and the similarity ratio becomes very high. However, there is no exact value of μ that can produce optimum results. Making μ adaptive and dynamic is essential in order to keep the update ratio as high as possible and similarity ratio as small as possible.

3.3 Multi-step Length

The step length *l* used in MWO to update the mussel position. The original adapted MWO uses one step length to update the ANN weights of input to hidden layers and hidden to output layers. The E-MWO uses separate steps length for each layer to layer update (e.g. first step length l_1 used in update process of weights input to hidden, second step length l_2 used in update process of weights hidden to output). Multi-step length will make the mussels' population more diverse, which will make the mussels' population more explorative [23]. Maintaining the population in balanced degree of diversity is essential to ensure that the solution space is adequately searched [24].

3.4 Dynamic Termination Criterion

The termination criterion for E-MWO depends on the dynamic quality measure in addition to the number of iterations reached. Two newly introduced quality measure; Update ratio U_R and similarity ratio S_R are utilized to guide the E-MWO when to terminate. Whereas U_R should not be less than ε_1, while S_R should not exceed ε_2 (i.e. $U_R > \varepsilon_1$, $S_R < \varepsilon_2$). S_R and U_R are explained earlier in equations (8) and (9), respectively. As these two relations are preserved as the solution is converging, thus better performance accuracy will be achieved.

3.5 The E-MWO-Based Training of ANN

The Feed-forward ANN weights (including biases) are adjusted and tuned using the E-MWO algorithm in order to solve a given classification problem as illustrated in Fig.1. Each network layer contains a number of neurons, which obtain their inputs from previous neurons-layer and forward the output to their following neurons-layer.

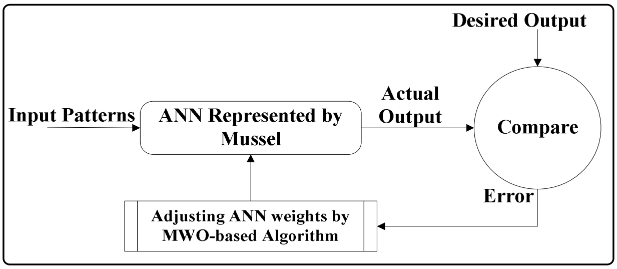

Desired Output
Input Patterns
ANN Represented by Mussel
Actual Output
Compare
Error
Adjusting ANN weights by MWO-based Algorithm

Fig. 1. Schematic diagram of MWO-based training of ANN

The Feed-forward ANN is represented by Vector-based scheme [5], [17], [21], [22]. Accordingly, each ANN is represented by a vector. This vector form the complete set of ANN structure with their corresponding weights and biases. Each mussel individual represents a complete feed-forward ANN. The sum squared errors (SSE) is considered as the objective function to evaluate the mussel fitness, i.e. minimizing the SSE. Mussel fitness and SSE described in equations (11) and (12), respectively. The bipolar-sigmoid is considered as the neuron activation function.

$$\text{Mussel fitness} = \frac{1}{SSE} \tag{11}$$

$$\boldsymbol{SSE} = \sum_{\boldsymbol{p=1}}^{\boldsymbol{NP}} \sum_{\boldsymbol{i=1}}^{\boldsymbol{NO}} \left(\boldsymbol{d_i^p} - \boldsymbol{y_i^p}\right)^{\boldsymbol{2}} \tag{12}$$

Where: *NP*: the number of patterns in the training dataset. *NO*: number of output neurons in output layer, d_i^p:the desired i^{th} output of p^{th} pattern, y_i^p : the actual i^{th} output of p^{th} pattern.

4 Results and Discussions

The E-MWO algorithms will be evaluated using four widely-used benchmarking classification datasets. The datasets obtained from UCI Machine Learning Repository [25], namely, Iris, Glass, Wisconsin Breast Cancer and Diabetes. The ANN structure was designed based on 3-layer architecture (input-hidden-output) as follows Iris:4-5-3, Cancer: 9-8-2, Diabetes: 8-7-2, Glass: 9-12-6. These ANN structures based on [14].

Each experimental session was conducted 20 times, whereas the mean and best-out of 20 values are reported for each algorithm. The coefficients and parameters' values of the MWO algorithm were set during the initialization step. These values are summarized as follows: Number of mussels (population size) N=100, the short-range reference α = 1.1, the long-range reference β=7.5, space scale-factor δ =20, the moving coefficients a = 1.1, b = 1.26, and c = 1.05, the walk scale factor γ = 0.15. Finally, the termination condition is the number of iterations G which is set to 200. These coefficients and parameters are similar as in [3] and [5]. Except the value of a is set to 1.1 to increase the explorative ability of the MWO, this is based on empirical experiments. The same parameter values that used in MWO are also used in E-MWO algorithm, while the newly introduced coefficients values of the E-MWO were set during the initialization step as follows: μ_c =1.5, λ_1= -0.2, λ_2=0.5, ε_1=0.2 andε_2= 0.8.

The new proposed E-MWO algorithm is compared against MWO [5], Harmony Search with Best-to-Worst ratio (HS-BtW) [14], and GA as presented by Dorsey et al. [27] from the category of evolutionary algorithms and BP from the category of gradient-based algorithms. All algorithms were implemented using Java and all tests were run on the same computer.

Different values of μ were used in MWO to explore its effect. Best classification accuracy were reported for MWO when μ=1.6 or μ=1.7 as shown in Fig. 2.

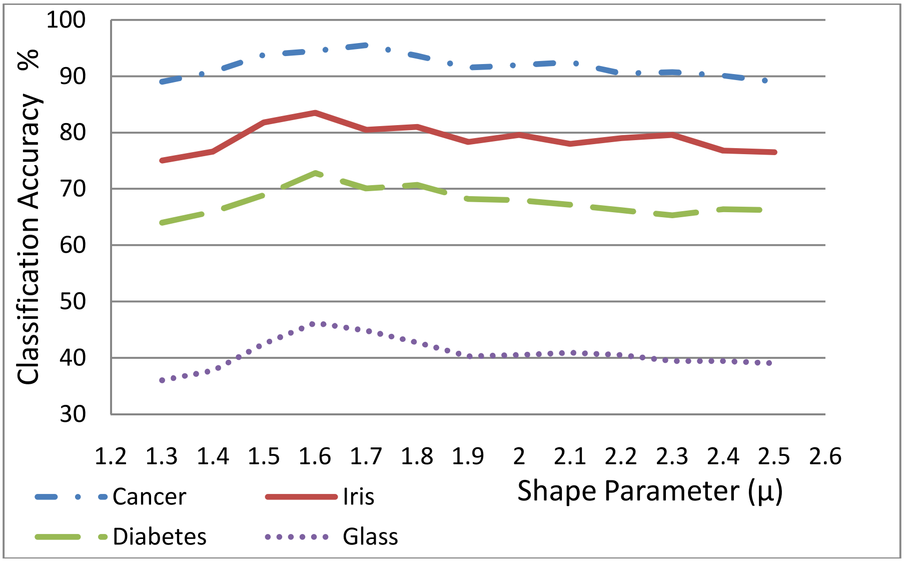


Fig. 2. Classification accuracy achieved by different values of μ

Clearly, results show that the E-MWO algorithm ranked first in achieving best classification accuracy in three datasets: Iris, Diabetes and Glass, while the HS-BtW algorithm scores best in cancer dataset problem as shown in Table 1. In terms of training time, the E-MWO and MWO consumes less training time with small difference than other algorithms, while BP records the large training time. Table 2 shows the training time of ANN by E-MWO-based training and other evolutionary and gradient based algorithms.

Table 1. Classification accuracy of ANN for different datasets

Dataset		**E-MWO**	**MWO**	**HS-BtW**	**GA**	**BP**
Iris	Best	**100.0**	**100**	96.6	96.6	96.6
	Mean	**91.0**	89.6	86.8	84.6	96.6
Cancer	Best	98.5	98.5	**99.2**	99.2	97.8
	Mean	97.1	97.3	**98.2**	97.4	96.1
Diabetes	Best	**92.8**	79.0	77.9	79.2	79.2
	Mean	**78.0**	74.5	75.3	73.8	75.4
Glass	Best	**95.3**	60.4	72.0	62.7	72.0
	Mean	58.7	49.1	58.8	45.2	**60.1**

Table 2. Training time of ANN for different datasets (time in seconds)

Dataset		**E-MWO**	**MWO**	**HS-BtW**	**GA**	**BP**
Iris	Best	**6.0**	5.0	49.0	10.0	1132.0
	Mean	**2.7**	4.4	58.3	8.0	826.4
Cancer	Best	**26.0**	30.0	30.0	1355.0	3909.0
	Mean	**25.9**	29.9	64.9	1349.4	4097.3
Diabetes	Best	**16.0**	26.0	49.0	175.0	16311.0
	Mean	**27.1**	25.8	72.7	392.3	14112.0
Glass	Best	20.0	**18.0**	64.0	740.0	6104.0
	Mean	**18.1**	18.7	69.1	639.7	3003.1

5 Conclusions

In this paper, an enhanced version of MWO is proposed; E-MWO. The E-MWO adaptively and dynamically sets the value of the shape parameter. It has multi-step length and new selection scheme to update the mussels' population positions. The termination criterion depends on two new dynamic quality measures: similarity ratio and update ratio.

The Feed-forward artificial neural networks have been trained by adapting the MWO and E-MWO algorithms. The pattern classification problem of different datasets has been tackled in this research in order to verify and test the algorithms functionality and efficiency. Two criteria are considered in the performance evaluation process; overall training convergence time and classification accuracy.

The results showed that the E-MWO algorithm has been successfully adapted and applied to train feed-forward artificial neural networks. The obtained results indicate that the E-MWO and the MWO algorithm are competitive in terms of convergence time. On the other hand, the best classification accuracy was achieved by E-MWO in three out of four datasets.

The future work includes further analysis for the MWO algorithm, by focusing on premature convergence problem. In addition, detailed description of the E-MWO and how it overcome the drawbacks of the MWO with deep analysis using more dataset benchmark problems should be investigated, which are currently ongoing works by the authors.

Acknowledgment. This research is supported by UNIVERSITI SAINS MALAYSIA and has been funded by the Research University Cluster (RUC) grant titled by "Reconstruction of the Neural Micro circuitry or Reward-Controlled Learning in the Rat Hippocampus" (1001/PSKOM/8630022).

References

1. Ghosh-Dastidar, S., Adeli, H.: A new supervised learning algorithm for multiple spiking neural networks with application in epilepsy and seizure detection. J. Neural Networks 22, 1419–1431 (2009)
2. Wang, C.H., Lin, S.F.: Toward a New Three Layer Neural Network with Dynamical Optimal Training Performance. In: Proceedings IEEE International Conference on Systems, Man and Cybernetics, pp. 3101–3106 (2007)
3. An, J., Kang, Q., Wang, L., Wu, Q.: Mussels Wandering Optimization: An Ecologically Inspired Algorithm for Global Optimization. Cognitive Computation 5(2), 188–199 (2013)
4. Suraweera, N.P., Ranasinghe, D.N.: A Natural Algorithmic Approach to the Structural Optimisation of Neural Networks. In: Proceedings of 4th International Conference on Information and Automation for Sustainability, pp. 150–156 (2008)
5. Abusnaina, A.A., Abdullah, R.: Mussels Wandering Optimization Algorithm based training of Artificial Neural Networks for Pattern Classification. In: Proceedings of the 4th International Conference on Computing and Informatics, ICOCI 2013, pp. 78–85 (2013)
6. Seiffert, U.: Training of Large-Scale FeedForward Neural Networks. In: Proc. of International Joint Conference on Neural Networks, Vancouver, BC, Canada (2006)
7. Jiang, X., Wah, A.H.K.S.: Constructing and training feed-forwardneural networks for pattern classification. Pattern Recognition 36, 853–867 (2003)
8. Baptista, D., Morgado-Dias, F.: A survey of artificial neural network training tools. Neural Computing and Applications 23(3-4), 609–615 (2013)
9. Afshar, S., Mosleh, M., Kheyrandish, M.: Presenting anew multiclass classifier based on learning automata. J. Neurocomputing 104, 97–104 (2013)
10. Zhang, G.P.: Neural Networks for Classification: A Survey. J. IEEE Transactions on Systems, Man, And Cybernetics–Part C: Applications And Reviews 30(4), 451–462 (2000)
11. Wu, T.K., Hwang, S.C., Meng, Y.R.: Improving ANN Classification Accuracy for the Identification of Students with LDs through Evolutionary Computation. In: Proceedings of the 2007 IEEE Congress on Evolutionary Computation, pp. 4358–4364 (2007)

12. Silva, D.N.G., Pacifico, L.D.S., Ludermir, T.B.: An evolutionary extreme learning machine based on group search optimization. In: IEEE Congress of Evolutionary Computation, pp. 574–580 (2011)
13. Huang, G.B., Zhu, Q.Y., Siew, C.K.: Extreme learning machine: A new learning scheme of feedforward neural networks. In: Proceedings of IEEE International Conference on Neural Networks, pp. 985–990 (2004)
14. Kattan, A., Abdullah, R.: Training of Feed-Forward Neural Networks for Pattern-Classification Applications Using Music Inspired Algorithm. International Journal of Computer Science and Information Security 9(11), 44–57 (2011)
15. Karaboga, D., Akay, B., Ozturk, C.: Artificial Bee Colony (ABC) Optimization Algorithm for Training Feed-Forward Neural Networks. In: Torra, V., Narukawa, Y., Yoshida, Y. (eds.) MDAI 2007. LNCS (LNAI), vol. 4617, pp. 318–329. Springer, Heidelberg (2007)
16. Gao, Q., Lei, K.Q.Y., He, Z.: An Improved Genetic Algorithm and Its Application in Artificial Neural Network. In: Proceedings Fifth International Conference on Information, Communications and Signal Processing, pp. 357–360 (2005)
17. He, S., Wu, Q.H., Saunders, J.R.: Group Search Optimizer: An Optimization Algorithm Inspired by Animal Searching Behavior. J. IEEE Transactions on Evolutionary Computation 13(5), 973–990 (2009)
18. He, S., Wu, Q.H., Saunders, J.R.: Breast cancer diagnosis using an artificial neural network trained by group search optimizer. J. Transactions of the Institute of Measurement and Control 31(6), 517–553 (2009)
19. Su, T., Jhang, J., Hou, C.: A hybrid Artificial Neural Networks and Particle Swarm Optimization for function approximation. International Journal of Innovative Computing, Information and Control 4(9), 2363–2374 (2008)
20. Kattan, A., Abdullah, R.: Training Feed-Forward Artificial Neural Networks For Pattern-Classification Using The Harmony Search Algorithm. In: Proceedings the Second International Conference on Digital Enterprise and Information Systems (DEIS 2013), pp. 84–97 (2013)
21. Kattan, A., Abdullah, R., Salam, R.: Harmony Search Based Supervised Training of Artificial Neural Networks. In: Proceedings International Conference on Intelligent Systems, Modeling and Simulation, pp. 105–110 (2010)
22. Fish, K., Johnson, J., Dorsey, E., Blodgett, J.: Using an Artificial Neural Network Trained with a Genetic Algorithm to Model Brand Share. Journal of Business Research 57, 79–85 (2004)
23. Zaharie, D.: Control of population diversity and adaptation in differential evolution algorithms. In: Matousek, R., Osmera, P. (eds.) Proceedings Mendel 9th International Conference Soft Computing, Brno, Czech Republic, pp. 41–46 (June 2003)
24. Gupta, D., Ghafir, S.: An Overview of methods maintaining Diversity in Genetic Algorithms. International Journal of Emerging Technology and Advanced Engineering 2(5), 56–60 (2012)
25. Frank, A., Asuncion, A.: UCI Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences, `http://archive.ics.uci.edu/ml` (online accessed on February 2013)
26. Masmoudi, M.S., Klabi, I., Masmoudi, M.: Performances improvement of back propagation algorithm applied to a lane following system. In: Proceedings World Congress on Computer and Information Technology, pp. 1–5 (2013)
27. Dorsey, R.E., Johnson, J.D., Mayer, W.J.: A Genetic Algoirthm for the Training of Feedforward Neural Networks. In: Advances in Artificial Intelligence in Economics, Finance, and Management, vol. 1, pp. 93–111 (1994)

Appendix A: The list of equations used in MWO algorithm [3]

Eq. No.	Formula	Parameters Description
Eq.(1)	$D_{ij} := \lVert x_i - x_j \rVert$ $= \left[\sum_{k=1}^{d} (x_{ik} - x_{jk})^2\right]^{1/2} \quad i, j \in \mathrm{N}_N$	D_{ij} : spatial distance between mussels m_i and m_j in S^d. N : number of mussels.
Eq. (2)	$\begin{cases} r_s(t) := \alpha \, . \, max_{\, i,j \, \in N}\{D_{ij}(t)\}/\delta \\ r_l(t) := \beta \, . \, max_{\, i,j \, \in N}\{D_{ij}(t)\}/\ \delta \end{cases}$	r_s : short-range reference. r_l : long-range reference. αandβ are positive constant coefficients with $\alpha<\beta$. $max_{i,j \in N}\{D_{ij}(t)\}$ Maximum distance among all mussels at iteration t. : scale factor of space, which depends on the problem to be solved.
Eq. (3)	$\zeta_{si} := \#(D_i < r_s) \,/\, (r_s N)$	ζ_{si} :short-range density, ζ_{li} : long-range density. Where $\#(A < b)$ is used to compute the count in set A satisfying $a<b$; $a \in$A; D_i is the distance matrix from mussel m_i to other mussels in the population.
Eq. (4)	$\zeta_{li} := \#(D_i < r_l) \,/\, (r_l N)$	
Eq. (5)	$P_i := \begin{cases} 1, \text{ if } a - b\zeta_{si} + c\zeta_{li} > z \\ 0, \ otherwise \end{cases}$	a, b, and $c$are positive constant coefficients. Z is a value randomly sampled from the uniform distribution [0,1].
Eq. (6)	$\ell_i := \gamma \, [1 - \mathrm{rand}()]^{-1/(\mu-1)}$	l_i : step length , μ : is the shape parameter, which it is known as the Le´vy exponent or scaling exponent that determines the movement strategy; $1.0 < \mu < 3.0$. γ : the walk scale factor.
Eq. (7)	$\grave{x}_i := \begin{cases} x_\mathrm{i} + \ell_i \Delta_\mathrm{g}, & if \ P_i = 1 \\ x_\mathrm{i}, & if \ \ P_i = 0 \end{cases}$	$\grave{x}_i$: new mussel-position coordinate. x_i:current mussel-position coordinate. x_g : best mussel-position coordinate. $\Delta_\mathrm{g} = x_i - x_g$.

Fuzzy Modified Great Deluge Algorithm for Attribute Reduction

Majdi Mafarja[1] and Salwani Abdullah[2]

[1] Department of Computer Science, Faculty of Information Technology
Birzeit University, Palestine
mmafarja@birzeit.edu
[2] Data Mining and Optimization Research Group (DMO),
Center for Artificial Intelligence Technology, Universiti Kebangsaan Malaysia
salwani@ftsm.ukm.my

Abstract. This paper proposes a local search meta-heuristic free of parameter tuning to solve the attribute reduction problem. Attribute reduction can be defined as the process of finding minimal subset of attributes from an original set with minimum loss of information. Rough set theory has been used for attribute reduction with much success. However, the reduction method inside rough set theory is applicable only to small datasets, since finding all possible reducts is a time consuming process. This motivates many researchers to find alternative approaches to solve the attribute reduction problem. The proposed method, Fuzzy Modified Great Deluge algorithm (Fuzzy-mGD), has one generic parameter which is controlled throughout the search process by using a fuzzy logic controller. Computational experiments confirmed that the Fuzzy-mGD algorithm produces good results, with greater efficiency for attribute reduction, when compared with other meta-heuristic approaches from the literature.

Keywords: Great Deluge, Fuzzy Logic, Attribute Reduction.

1 Introduction

Attribute Reduction (AR) which is a NP-hard problem [1] can be defined as the problem of finding minimal attributes (subset) from the original set of features. It has become a necessary pre-processing step to reduce the complexity of data mining process by removing the irrelevant and/or redundant attributes. Recently, many researchers tried to implement the stochastic methods to solve attribute reduction problem such as tabu search [2, 3], ant colony optimisation (AntRSAR) [4], genetic algorithm (GenRSAR) [5, 6], simulated annealing (SimRSAR) [6], ant colony optimisation (ACOAR) [7-11], scatter search (SSAR) [12, 13]), great deluge algorithm (GD-RSAR) [14], composite neighbourhood structure (IS-CNS) [15], hybrid variable neighbourhood search algorithm (HVNS-AR) [16], and a constructive hyper-heuristics (CHH_RSAR) [17].

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_19, © Springer International Publishing Switzerland 2014

Great Deluge algorithm (GD) [18] is one of the more recent meta-heuristics originally developed as a variant of simulated annealing algorithm. It is a local search procedure that allows worse solutions to be accepted based on some given lower boundary or "*level*". A modified great deluge algorithm (called m-GD), proposed by Mafarja and Abdullah [19] , uses an intelligent mechanism to control the increasing rate (β) of the *"level"* instead of using the linear mechanism used in the original great deluge algorithm. In m-GD the search space is divided into three regions of equal size. The *level* is updated using different increasing rate β according to the region that the level belongs to. This paper proposed an enhancement on the former approach, where a fuzzy logic controller is used to control the value of the single parameter in the algorithm in order to achieve the best possible performance of the algorithm. This approach is called fuzzy modified great deluge for attribute reduction (Fuzzy-mGD).

The paper is structured as follows: Section 2 introduces the proposed fuzzy modified great deluge approach for attribute reduction problem. Section 3 reports the experimental results on the attribute reduction problem. This paper ends with a conclusion and a short summary of our results in Section 4.

2 Great Delude Algorithm (GD)

Great Deluge algorithm (GD) which was originally proposed by Dueck [18] is a generic algorithm applied to optimization problems. It is a local search procedure that allows worse solutions to be accepted based on some given lower boundary or "*level*". The general pseudo code for the great deluge algorithm is shown in Fig. 1. GD is a variant of simulated annealing algorithm (SA) with a different acceptance mechanism for accepting non-improving solution. It depends only on one parameter which is the increasing rate (β) of the water level [18].

```
Input: level L.
s = $s_0$ ; /* Generation of the initial solution */
Choose the rain speed $\beta$; /* $\beta > 0$ */
Choose the initial water level level;
Repeat
    Generate a random neighbor $s'$ ;
    If $f(s') < level$ Then $s = s'$ /* Accept the neighbor solution */
    $level = level - \beta$ ; /* update the water level */
Until Stopping criteria satisfied
Output: Best solution found
```

Fig. 1. A general GD algorithm pseudo code adopted from [20]

GD algorithm always accept a better solution, a worse solution is accepted if the quality of the solution is less than (for minimisation problems) or equal to some given upper boundary value which is called a "*level*". The "*level*" is initially set to be the objective function value of the initial solution, and is iteratively increased by a constant β (where β is referred as an increasing rate in this work) during its run.

3 Fuzzy Logic Controller

Fuzzy Logic has been widely used with many real world applications since being introduced by Zadeh in 1965 [21]. For example, Jensen and Shen [22] have proposed three new techniques for fuzzy rough set feature selection based on the use of fuzzy T-transitive similarity relations. Also in scheduling and timetabling applications, fuzzy evaluation functions have been utilised in a number of different applications.

The fuzzy systems are generally consist of four components; an input fuzzifier, a knowledge base (rule base), an interfaces engine and defuzzification inference (see Fig. 2). The rules have a main role of linking the input and output variables (in `IF - THEN' form) are utilised to depict the response of the system relatively in terms of linguistic variables (words) than the mathematical formulae (see Table 1).

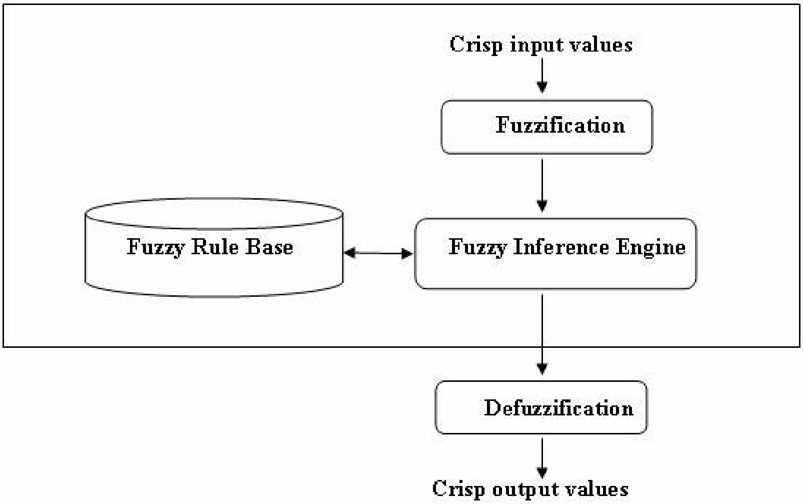


Fig. 2. Structure of a Fuzzy Logic Model

The `IF' part of the rule is mentioned as the `antecedent' and the `THEN' part is mentioned as the `consequent'. The number of inputs and outputs and as well as the desired behaviour of the system have direct impact on the number of rules. After the rules are generated, the system can be seen as a non-linear mapping from inputs to outputs. More details about simple treatment can be found in Cox [23] and complete treatment in Zimmerman [24].

4 Fuzzy Modified Great Deluge for Attribute Reduction (Fuzzy-mGD)

A fuzzy logic controller is used to control the increasing rate (β) parameter value intelligently, based on the quality of the produced solutions during the searching processes.

4.1 Solution Representation and Initial Solution Generation

In this work, a solution is represented in one dimensional vector, where the length of the vector is based on the number of attributes of the original dataset. Each value in the vector (cell) is represented by “1” or “0”. Value “1” shows that the corresponding attribute is selected; otherwise the value is set to “0”. Fig. 3 shows the subset of the solution where 4 attributes are selected.

0	1	0	0	0	1	0	0	1	0	1

Fig. 3. Representation of the solution

4.2 Neighbourhood Structure

In this work, the neighbourhood of a trial solution (called Sol_{trial}) is generated by a random flip-flop where three cells are selected at random from the current solution (*Sol*). For each selected cell, if its value is “1” then it is changed to “0”, which means that the feature is deleted from the current solution. Otherwise, it is added by changing “0” to “1”. The cardinality of the generated trial solution must not exceed the cardinality of the best solution so far.

4.3 Quality Measure

The quality of the solutions is measured based on the dependency degree (calculated based on a rough set theory (RST) [25]. Two given solutions: best solution, Sol_{best}, and trial solution, Sol_{trial}. The trial solution Sol_{trial} is accepted if there is an improvement in the dependency degree i.e. ($f(Sol_{trial}) > f(Sol_{best})$). However if the dependency degree for both solutions is the same, the solution with the less cardinality is accepted.

4.4 The Algorithm

In this work, we consider the Fuzzy-mGD as a MISO (Multi Input Single Output) dynamical system; by sampling the Fuzzy-mGD outputs and acting on its inputs according to the fuzzy rules. By using the fuzzy controller, the *level* is updated by applying different values through the search process instead of using one increasing rate (as in the original GD), or three increasing rates (as in [19]). The search space is divided into three equalled areas; each one represents a fuzzy set (*low*, *medium* and *high*) in the fuzzy logic system as shown in Fig. 4. The controller takes two inputs, the trial solution (Sol_{trial}) and the best solution (Sol_{best}), that are connected to the general terms: *low*, *medium* and *high* (corresponding to fuzzy sets meanings). A set rules that links the input variables (Sol_{trial} and Sol_{best}) with the single output variable (β), is built according to the fuzzy rules in Table 1.

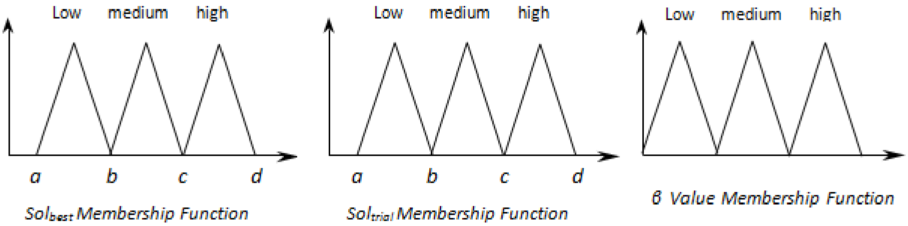

Low
medium
high
a
b
c
d
Sol_{best} Membership Function
Low
medium
high
a
b
c
d
Sol_{trial} Membership Function
Low
medium
high
β Value Membership Function

Fig. 4. Graphical representation of the membership functions of Fuzzy-mGD

For example, when $f(Sol_{best})$ is “Low” and $f(Sol_{trial})$ is “Low”, it means that both solutions fall in the low fuzzy set, and the *level* will be updated according to the degree of membership of an input value to the fuzzy set.

Table 1. The membership functions distribution

		$f(Sol_{best})$		
		Low	**Medium**	**High**
$f(Sol_{trial})$	**Low**	Low	Low	Medium
	Medium	Low	Medium	Medium
	High	Medium	High	High

Listed below are the typical control rules that are used to exemplify the performance of this fuzzy system.

R_1 IF (Sol_{trial} is *low*) AND (Sol_{best} is *low*) THEN (β is *low*)

R_2 IF (Sol_{trial} is *low*) AND (Sol_{best} is *medium*) THEN (β is *low*)

R_3 IF (Sol_{trial} is *low*) AND (Sol_{best} is *high*) THEN (β is *medium*)

R_4 IF (Sol_{trial} is *medium*) AND (Sol_{best} is *low*) THEN (β is *low*)

R_5 IF (Sol_{trial} is *medium*) AND (Sol_{best} is *medium*) THEN (β is *medium*)

R_6 IF (Sol_{trial} is *medium*) AND (Sol_{best} is *high*) THEN (β is *medium*)

R_7 IF (Sol_{trial} is *high*) AND (Sol_{best} is *low*) THEN (β is *medium*)

R_8 IF (Sol_{trial} is *high*) AND (Sol_{best} is *medium*) THEN (β is *high*)

R_9 IF (Sol_{trial} is *high*) AND (Sol_{best} is *high*) THEN (β is *high*)

For each of these inputs and output, three symmetric and triangular-shaped membership functions are defined and evenly distributed on the appropriate universe of discourse. A membership function gives the degree of membership of an input value to every fuzzy set as in Fig. 2, where *a* is the quality of the initial solution and *d* equal 1 (the maximum dependency degree). The input may belong to more than one fuzzy set. Depending on the membership functions, the `fuzzifier' calculates the grade of membership of each input variable for every rule. For example, in R_2, the membership grade is calculated for the Sol_{trial} in the fuzzy set *low* and for the Sol_{best} in the *medium* fuzzy set. The result represents β value that is used as the input values for Sol_{best} and Sol_{trial}.

5 Experimental Results

This section presents the results of the experimental studies using the proposed approach. The proposed algorithm was programmed using J2EE Java and performed on an Intel Pentium 4, 2.33 GHz computer, and tested on 13 well-known UCI datasets [6]. For every dataset, the algorithm was executed for 20 times. The comparisons are carried out in terms of the minimal attributes. The purpose of this comparison is to evaluate the effectiveness of using the fuzzy logic controller (as an intelligent mechanism to control the value of the parameter in each algorithm) in obtaining the minimal attributes.The superscripts in parentheses represent the number of runs that achieved this number of attributes, while the number of attributes without superscripts means that the method could obtain that number of attributes in all of the runs.

Table 2 and Table 3 show the minimal reducts that were obtained by Fuzzy-mGD and the state-of-art approaches. The methods in comparison are as follows:

- Simulated annealing (SimRSAR) by Jensen and Shen [6]
- Tabu search (TSAR) by Hedar et al. [2]
- Great deluge algorithm (GD-RSAR) by Abdullah and Jaddi [14]
- Composite neighbourhood structure (IS-CNS) by Jihad and Abdullah [15]
- Hybrid variable neighbourhood searchalgorithm (HVNS-AR) by Arajy and Abdullah [16]
- Constructive hyper-heuristics (CHH_RSAR) by Abdullah et al. [17].
- Ant colony optimisation (AntRSAR) by Jensen and Shen [4, 6]
- Genetic algorithm (GenRSAR) by Jensen and Shen [4, 6]
- Ant colony optimisation (ACOAR) by Ke et al. [8]
- Scatter search (SSAR) by Jue et al. [26]

Based on the results presented in Table 2 and Table 3, it can be seen that Fuzzy-mGD is comparable with the other approaches since it performs better than most of them. It is better than AntRSAR on five datasets, and better than SSAR on six datasets (ties on five datasets). Our approach is able to produce better results in all datasets when compared with GenRSAR method. Fuzzy-mGD too, has obtained

Table 2. Results of the experiments compared with those in literature 1

Datasets	Fuzzy-mGD	GD-RSAR	TSAR	SimRSAR	AntRSAR	ACOAR
M-of-N	6	$6^{(10)}\ 7^{(10)}$	6	6	6	6
Exactly	6	$6^{(7)}\ 7^{(10)}8^{(3)}$	6	6	6	6
Exactly2	10	$10^{(14)}11^{(6)}$	10	10	10	10
Heart	$6^{(9)}\ 7^{(11)}$	$9^{(4)}10^{(16)}$	6	$6^{(29)}\ 7^{(1)}$	$6^{(18)}\ 7^{(2)}$	6
Vote	8	$9^{(17)}10^{(3)}$	8	$8^{(15)}\ 9^{(15)}$	8	8
Credit	$8^{(18)}\ 9^{(2)}$	$11^{(11)}12^{(9)}$	$8^{(13)}\ 9^{(5)}\ 10^{(2)}$	$8^{(18)}\ 9^{(1)}\ 11^{(1)}$	$8^{(12)}\ 9^{(4)}\ 10^{(4)}$	$8^{(16)}9^{(4)}$
Mushroom	4	$4^{(8)}\ 5^{(9)}6^{(3)}$	$4^{(17)}\ 5^{(3)}$	4	4	4
LED	5	$8^{(14)}9^{(6)}$	5	5	$5^{(12)}\ 6^{(4)}\ 7^{(3)}$	5
Letters	8	$8^{(7)}9^{(13)}$	$8^{(17)}\ 9^{(3)}$	8	8	8
Derm	$6^{(19)}\ 8^{(1)}$	$12^{(14)}13^{(6)}$	$6^{(14)}\ 7^{(6)}$	$6^{(12)}\ 7^{(8)}$	$6^{(17)}\ 7^{(3)}$	6
Derm2	$8^{(7)}\ 9^{(13)}$	$11^{(14)}12^{(6)}$	$8^{(2)}\ 9^{(14)}\ 10^{(4)}$	$8^{(3)}\ 9^{(7)}$	$8^{(3)}\ 9^{(17)}$	$8^{(4)}9^{(16)}$
WQ	$12^{(5)}\ 13^{(14)}\ 14^{(1)}$	$15^{(14)}16^{(6)}$	$12^{(1)}\ 13^{(13)}\ 14^{(6)}$	$13^{(16)}\ 14^{(4)}$	$12^{(2)}\ 13^{(7)}\ 14^{(11)}$	$12^{(4)}13^{(12)}14^{(4)}$
Lung	$4^{(15)}\ 5^{(5)}$	$4^{(5)}\ 5^{(2)}\ 6^{(13)}$	$4^{(6)}\ 5^{(13)}\ 6^{(1)}$	$4^{(7)}\ 5^{(12)}\ 6^{(1)}$	4	4

Table 3. Results of the experiments compared with those in literature 2

Datasets	Fuzzy-mGD	IS-CNS	HVNS-AR	GenRSAR	CHH_RSAR	SSAR
M-of-N	6	6	6	$6^{(6)}7^{(12)}$	$6^{(11)}7^{(9)}$	6
Exactly	6	6	6	$6^{(10)}7^{(10)}$	$6^{(13)}7^{(7)}$	6
Exactly2	10	10	10	$10^{(9)}11^{(11)}$	10	10
Heart	$6^{(9)}\ 7^{(11)}$	6	6	$6^{(18)}7^{(2)}$	6	6
Vote	8	8	8	$8^{(2)}9^{(18)}$	8	8
Credit	$8^{(18)}\ 9^{(2)}$	$8^{(10)}9^{(9)}\ 10^{(1)}$	$8^{(7)}9^{(6)}\ 10^{(7)}$	$10^{(6)}11^{(14)}$	$8^{(10)}9^{(7)}\ 10^{(3)}$	$8^{(9)}\ 9^{(8)}\ 10^{(3)}$
Mushroom	4	4	4	$5^{(1)}6^{(5)}7^{(14)}$	4	$4^{(12)}\ 5^{(8)}$
LED	5	5	5	$6^{(1)}7^{(3)}8^{(16)}$	5	5
Letters	8	8	8	$8^{(8)}9^{(12)}$	8	$8^{(5)}\ 9^{(15)}$
Derm	$6^{(19)}\ 8^{(1)}$	$6^{(18)}\ 7^{(2)}$	$6^{(16)}\ 7^{(4)}$	$10^{(6)}11^{(14)}$	6	6
Derm2	$8^{(7)}\ 9^{(13)}$	$8^{(4)}9^{(16)}$	$8^{(5)}9^{(12)}10^{(3)}$	$10^{(4)}11^{(16)}$	$8^{(5)}9^{(5)}10^{(10)}$	$8^{(2)}\ 9^{(18)}$
WQ	$12^{(5)}\ 13^{(14)}\ 14^{(1)}$	$12^{(2)}13^{(8)}14^{(10)}$	$12^{(3)}13^{(6)}14^{(8)}\ 15^{(3)}$	16	$12^{(13)}14^{(7)}$	$13^{(4)}\ 14^{(16)}$
Lung	$4^{(15)}\ 5^{(5)}$	$4^{(17)}\ 5^{(3)}$	$4^{(16)}\ 5^{(4)}$	$6^{(8)}7^{(12)}$	$4^{(10)}\ 5^{(7)}\ 6^{(3)}$	4

better results than SimRSAR in six datasets and TSAR in eight datasets. The proposed Fuzzy-mGD is able to obtain better results on all datasets when compared with the GD-RSAR. It can produce better results than IS-CNS, HVNS-AR, CHH_RSAR in 6, 5, and 7 instances, respectively. Fuzzy-mGD is able to obtain two results better than ACOAR. In general, we can summarise that our approach is better than most of the approaches introduced. Fuzzy-mGD demonstrates highly promising performance when compared with other available methods. We believe that the strength of the method comes from the improvement of the new modification on the GD algorithm that embeds the fuzzy logic controller to control the parameter β which further enhanced the performance of the proposed approach through a better exploitation during the search process.

6 Conclusions

The work described in this paper proposed a fuzzy modified great deluge algorithm, called Fuzzy-mGD, to solve the attribute reduction problem in the rough set theory. Great Deluge algorithm has only one generic parameter which is controlled throughout the search process using a fuzzy logic controller by taking into account the quality of the produced solutions. Several benchmark UCI datasets are used to evaluate the utilisation efficiency of the proposed method. The experimental results showed that our approach provides qualified solutions to the well-known benchmark datasets from the attribute reduction literature. Employing a fuzzy logic controller positively influences the performance of the original algorithm by producing a lower number of minimal attributes. As a result, we can say that controlling the parameter values affects the behaviour of the Fuzzy-mGD method in searching for the most informative attributes and that the selected subset of attributes is a better representation of the original data.

References

1. Skowron, A., Rauszer, C.: The discernibility matrices and functions in information systems. In: Slowinski, R. (ed.) Intelligent Decision Support–Handbook of Applications and Advances of the Rough Set Theory, pp. 311–362. Kluwer Academic Publishers, Poland (1992)
2. Hedar, A.-R., Wang, J., Fukushima, M.: Tabu search for attribute reduction in rough set theory. Soft. Comput. 12, 909–918 (2008)
3. Wang, J., Guo, K., Wang, S.: Rough set and Tabu search based feature selection for credit scoring. Procedia Computer Science 1, 2425–2432 (2010)
4. Jensen, R., Shen, Q.: Finding Rough Set Reducts with Ant Colony Optimization. In: Proceedings of the 2003 UK Workshop on Computational Intelligence, pp. 15–22 (2003)
5. BingXiang, L., Feng, L., Xiang, C.: An adaptive genetic algorithm based on rough set attribute reduction. In: 2010 3rd International Conference on Biomedical Engineering and Informatics (BMEI), pp. 2880–2883 (2010)
6. Jensen, R., Shen, Q.: Semantics-Preserving Dimensionality Reduction: Rough and Fuzzy-Rough-Based Approaches. IEEE Trans. on Knowl. and Data Eng. 16, 1457–1471 (2004)
7. Wu, J., Qiu, T., Wang, L., Huang, H.: An Approach to Feature Selection Based on Ant Colony Optimization and Rough Set. In: Chen, R. (ed.) ICICIS 2011 Part I. CCIS, vol. 134, pp. 466–471. Springer, Heidelberg (2011)
8. Ke, L., Feng, Z., Ren, Z.: An efficient ant colony optimization approach to attribute reduction in rough set theory. Pattern Recog. Lett. 29, 1351–1357 (2008)

9. Chen, Y., Miao, D., Wang, R.: A rough set approach to feature selection based on ant colony optimization. Pattern Recog. Lett. 31, 226–233 (2010)
10. Ming, H.: Feature Selection Based on Ant Colony Optimization and Rough Set Theory. In: International Symposium on Computer Science and Computational Technology, ISCSCT 2008, vol. 1, pp. 247–250 (2008)
11. Wang, G., Wang, S.-J., Shi, L., Huang, D., Chen, H., Liu, Y., Peng, X.: Study of adaptive parameter control for ant colony optimization applied to feature selection problem. Advanced Science Letters (2012) (in press)
12. Jue, W., Hedar, A.R., Shouyang, W.: Scatter Search for Rough Set Attribute Reduction. In: Second International Conference on Bio-Inspired Computing: Theories and Applications, BIC-TA 2007, pp. 236–240 (2007)
13. Wang, J., Hedar, A.-R., Wang, S., Ma, J.: Rough set and scatter search metaheuristic based feature selection for credit scoring. Expert Systems with Applications 39, 6123–6128 (2012)
14. Abdullah, S., Jaddi, N.S.: Great Deluge Algorithm for Rough Set Attribute Reduction. In: Zhang, Y., Cuzzocrea, A., Ma, J., Chung, K.-i., Arslan, T., Song, X. (eds.) DTA and BSBT 2010. CCIS, vol. 118, pp. 189–197. Springer, Heidelberg (2010)
15. Jihad, S.K., Abdullah, S.: Investigating composite neighbourhood structure for attribute reduction in rough set theory. In: 10th International Conference on Intelligent Systems Design and Applications (ISDA), pp. 1015–1020 (2010)
16. Arajy, Y.Z., Abdullah, S.: Hybrid variable neighbourhood search algorithm for attribute reduction in Rough Set Theory. In: Intelligent Systems Design and Applications (ISDA), pp. 1015–1020 (2010)
17. Abdullah, S., Sabar, N.R., Nazri, M.Z.A., Turabieh, H., McCollum, B.: A constructive hyper-heuristics for rough set attribute reduction. In: Intelligent Systems Design and Applications (ISDA), pp. 1032–1035 (2010)
18. Dueck, G.: New Optimization Heuristics The Great Deluge Algorithm and the Record-to-Record Travel. Journal of Computational Physics 104, 86–92 (1993)
19. Mafarja, M., Abdullah, S.: Modified great deluge for attribute reduction in rough set theory. In: Fuzzy Systems and Knowledge Discovery (FSKD), vol. 3, pp. 1464–1469. IEEE (2011)
20. Talbi, E.G.: Metaheuristics From design to implementation. Wiley Online Library (2009)
21. Zadeh, L.A.: Fuzzy sets. Information and Control 8, 338–353 (1965)
22. Jensen, R., Shen, Q.: New approaches to fuzzy-rough feature selection. IEEE Transactions on Fuzzy Systems 17, 824–838 (2009)
23. Cox, E.: The fuzzy systems handbook: a practitioner's guide to building, using, and maintaining fuzzy systems. Academic Press Professional, Inc. (1994)
24. Zimmermann, H.: Fuzzy Set Theory and its Applications. Kluwer Academic Publishers, Boston (1996)
25. Pawlak, Z.: Rough Sets. International Journal of Information and Computer Sciences 11, 341–356 (1982)
26. Wang, J., Hedar, A.-R., Zheng, G., Wang, S.: Scatter Search for Rough Set Attribute Reduction, pp. 531–535. IEEE (2009)

Fuzzy Random Regression to Improve Coefficient Determination in Fuzzy Random Environment

Nureize Arbaiy and Hamijah Mohd Rahman

Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn, Parit Raja
86400 Batu Pahat, Johor
nureize@uthm.edu.my, hamijah_mee@yahoo.com

Abstract. Determining the coefficient value is important to measure relationship in algebraic expression and to build a mathematical model though it is complex and troublesome. Additionally, providing precise value for the coefficient is difficult when it deals with fuzzy information and the existence of random information increase the complexity of deciding the coefficient. Hence, this paper proposes a fuzzy random regression method to estimate the coefficient values for which statistical data contains simultaneous fuzzy random information. A numerical example illustrates the proposed solution approach whereby coefficient values are successfully deduced from the statistical data and the fuzziness and randomness were treated based on the property of fuzzy random regression. The implementation of the fuzzy random regression method shows the significant capabilities to estimate the coefficient value to further improve the model setting of production planning problem which retain simultaneous uncertainties.

Keywords: Coefficient, fuzzy random variable, fuzzy random regression.

1 Introduction

Coefficient value is a constant which represents the rate of change of one variable as a function of changes in the other. The coefficient value is used to explain the relation between variables. This coefficient value plays a pivotal role in the development of mathematical programming especially in linear programming where the value is required for each of the attribute that include in the model. However, determining coefficient sometimes are difficult if relevant data are not available or difficult to obtain though it is commonly decided by the decision maker. In fact, the decision maker might give imprecise and vague information and may vary from one decision maker to another.

Therefore, the statistical tool is needed to extract the coefficient value from the historical data. One possible method to solve the problem of model's coefficient estimation is by using regression analysis. This regression analysis is a statistical technique for investigating and modeling the relationship between variables [1]. This technique is supported by effective statistical analysis dealing with numeric crisp

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_20, © Springer International Publishing Switzerland 2014

data. Meaning that the obtained statistical model is not considering the imprecision or uncertainty in the data. However, in real life application, there exist uncertainty fuzzy data in which subjective human estimation play a main role. Therefore, a fuzzy regression models was introduced to cope with the fuzzy uncertainty input-output data.

In addition, the historical data which is captured from the real situation may contain simultaneous fuzzy random information. We explain this situation of fuzzy random with an example. Let us assume the natural rubber production where two type of rubber is produced which is dry and latex. Randomness occurs because we collect the rubber production data in different mills and it is not known which mill produce higher production. Concurrently, fuzziness is characterized when the observed production data may contain imprecision. The existence of randomness in the data is beyond the scope of fuzzy regression approach. Therefore, an appropriate method is necessary to estimate the coefficient value when the fuzzy random information coexists in the historical data. Motivated by the above-mentioned problems, this paper utilizes a fuzzy random regression method to estimate the model's coefficient value whereby fuzzy random information is captured in the data.

The remainder of this paper is organized as follows. Section 2 describes the preliminary studies of regression model. Section 3explains the fuzzy random regression method to determine the model coefficients. Section 4 explains the solution of numerical experiment. The result of experiment is Section 5. Section 6 draws conclusions.

2 Preliminary Studies

Predicting the future is a basic problem in which people have to solve every day. It is a component of planning, decision-making, memory and causal reasoning [2]. The ability to accurately predict the future outcomes of complex systems has been the goal of forecasters for decades [3]. Prediction is widely been used in real life application such as in weather forecasts, medical studies, quality testing and many more. In order to accomplish a prediction, an appropriate method is necessary as a tool assist the accurate to prediction. Several methods for prediction have been suggested and implemented such as Kalman filters, Artificial Neural Network and also regression analysis. Regression analysis is one of the technique uses in statistical data for coefficient estimation. This technique is supported by effective statistical analysis dealing with precise crisp data. The importance of this regression approach is to investigate and model the relationship between variables. Regression analysis is used in evaluating the functional relationship between the dependent and independent variables and also in determining the best-fit model for describing the relationship [4].

Regression based on fuzzy random data is widely been used and many researcher study this technique to solve problem with this hybrid data [5]. The study of this fuzzy regression analysis based on fuzzy random variables with confidence intervals in the fuzzy multi attribute decision making was been proposed, as to enable decision makers to evaluate and find the importance weight [6]. Determining weight in multi-attribute decision making is important, so we propose a fuzzy random regression to estimate attribute importance in total evaluation within the bound of hybrid uncertainty [7]. Moreover, the fuzzy random regression method also has been proposed as an integral component of regression models in handling the existence of fuzzy random information [7]. Fuzzy random regression is a regression technique that was proposed as to cope with the fuzzy random data. In real world regression analysis, statistical data may be linguistically imprecise or vague where the real data cannot be characterized by using only the formalism of random variables as there is existence of stochastic and fuzzy uncertainty [8]. Statistical inference with fuzzy random data transfers the fuzziness into parameter estimators and it may necessary to defuzzify the vague parameter at level decision making [9]. Fuzzy random variables are characterized by the expected value and confidence interval. The detail explanation of fuzzy random variable and fuzzy random regression are given elsewhere [8][10]. Fuzzy Random Regression Model utilized to estimate the coefficient values for which the statistical data used contain simultaneous fuzzy and random information. Confidence interval is used to explain fuzzy random in regression model.

The input and output data for fuzzy random regression is represented as triangular fuzzy number. Fuzzy random data Y_j (output) and X_{jk} (input) for all $j = 1,\dots,N$ and $\mathrm{k} = 1,\dots,\mathrm{K}$ are defined as $Y_j = \bigcup_{t=1}^{M_{Y_j}} \left\{ \left(Y_j^t, Y_j^{t,l}, Y_j^{t,r} \right)_{\Delta}, p_j^t \right\}$ and $X_{jk} = \bigcup_{t=1}^{M_{X_{jk}}} \left\{ \left(X_j^t, X_j^{t,l}, X_j^{t,r} \right)_{\Delta}, q_{jk}^t \right\}$ respectively. Fuzzy variables $\left(Y_j^t, Y_j^{t,l}, Y_j^{t,r} \right)_{\Delta}$ and $\left(X_j^t, X_j^{t,l}, X_j^{t,r} \right)_{\Delta}$ are obtained with probability p_j^t and q_{jk}^t for $j = 1,\dots,n$, $k = 1,\dots,K$ and $t = 1,\dots,M$ or $t = 1,\dots,M_{X_{jk}}$, respectively.

Therefore, the fuzzy regression model is represented as $\mathrm{Y}_\mathrm{j}^* = \mathrm{A}_\mathrm{j}^* X_{j1} + \dots + \mathrm{A}_\mathrm{K}^* X_{jK} \underset{FR}{\supset} Y_i$, $j = 1,\dots,n$ where $\underset{FR}{\supset}$ is a fuzzy random inclusion relation [8], and is solvable using linear program.

Hence, the fuzzy random regression model with $\sigma -$ confidence intervals [8] is described as follows:

$$\begin{aligned} &\underset{\mathrm{A}}{\mathbf{min}} \quad J(A) = \sum_{k=1}^{K} \left(A_k^r - A_k^l \right) \\ &A_k^r \geq A_k^l, \\ &\mathrm{Y}_\mathrm{j}^* = \mathrm{A}_\mathrm{j}^* I\left[e_{X_{j1}}, \sigma_{X_{j1}} \right] + \dots + \mathrm{A}_\mathrm{K}^* I\left[e_{X_{jK}}, \sigma_{X_{jK}} \right] \underset{\tilde{h}}{\supset} I\left[e_{Y_j}, \sigma_{Y_j} \right] \\ &j = 1,\dots,n;\ k = 1,\dots,K, \end{aligned} \tag{1}$$

where confidence interval $I\left[e_{X_{j1}}, \sigma_{X_{j1}} \right]$ that is induced by the expectation and variance of a fuzzy random variable is shown as follows:

$$I\left[e_X, \sigma_X \right] \underline{\underline{\Delta}} \left[E(X) - \sqrt{\mathrm{var}(X)}, E(X) + \sqrt{\mathrm{var}(X)} \right] \tag{2}$$

The model describe in (2) is then used to model the problem in consideration.

3 Solution Model

Fuzzy random regression framework to estimate the coefficient is as follows:

Step 1: Problem Description
Step 2: Data Acquisition and Preparation
Step 3: Fuzzy Random Regression Model for estimation
3.1 Eliciting the confidence interval
The confidence interval of each fuzzy random variable is computed by inducing the expected value and variance of fuzzy random variable to construct the one-sigma confidence interval, $I[e_X \ , \sigma_X \ ]$.

3.2 Estimating the coefficient
Fuzzy random regression analysis is used to estimate the coefficient. Let A_k denote an attribute for $k = 1,\ldots,K$ and Y_j^* is the total evaluation for $j = 1,\ldots,n$, where n is the number of candidate alternatives to be evaluated. A fuzzy random regression model (10) is described as follows:

$$\begin{aligned} \min_{A} \quad & J(A) = \sum_{k=1}^{K} \left(\bar{a}_k - \underline{a}_k \right) \\ \text{subject to}: \quad & \bar{a}_k \geq \underline{a}_k, \\ & y_j^r + (e_{Y_j} + \sigma_{Y_j}) \leq \sum_{k=1}^{K} \bar{a}_k \left(e_{X_{j1}} + \sigma_{X_{j1}} \right), \\ & y_j^l - (e_{Y_j} - \sigma_{Y_j}) \geq \sum_{k=1}^{K} \underline{a}_k \left(e_{X_{j1}} - \sigma_{X_{j1}} \right), \\ & j = 1,\ldots,n; \quad k = 1,\ldots,K \end{aligned} \tag{3}$$

Step 4: Decision-making and Analysis

The coefficients $[\underline{a}_k, \bar{a}_k]$ are obtained from fuzzy random regression model (3). The estimated coefficient deduced by fuzzy random regression can be used for several purposes such as to develop a production planning model and to calculate final score of evaluation for selecting the best samples among alternatives.

4 Numerical Experiments

We demonstrate the use of the proposed method on a rubber production in Malaysia. The natural rubber industry is one of Malaysia's economic contributors where this industry contributed almost RM36.4 billion in export earnings in 2012 [11]. The rubber products accounted for 3.9% of Malaysia's total exports for manufacturing products. More than 500 manufacturers included in the Malaysian rubber product industry which produces latex products, tyres and tyre-related products and also industrial and general rubber products [12]. In the latex products sub sector, 125 manufacturers are involving which produce gloves, condom, catheters and others. In fact, Malaysia is the world's leading producer and exporter of catheters, latex thread and natural rubber medical gloves [13].

The dataset of models of rubber production are tabulated in Table 1 respectively. Two main products are produced from the rubber plant, namely dry and latex. The two rubbers have different applications and market outlets; dry rubber is used mainly for general rubber product, while latex is used in producing gloves, catheters, latex thread and others. Thus, in this experiment the coefficient values are estimated for the two products of rubber, which are the decision factor of the model. The estimated value can further been use for setting the production planning model. However, in this study, we restrict ourselves to explain only the estimation of coefficient from statistical data which contain fuzzy random information.

4.1 Data Preparation for Coefficient Estimation

The following step involves the data preparation to estimate the decision coefficients by using a fuzzy random based regression model. Two data sets are collected from the Malaysian Rubber Board [14] for dry and latex production 13 years from 2000 to 2012. The derived input data, applied as decision variable coefficients in this study, were therefore based on the previous actual data for production. Sample of data preparation is shown in Tables 2 in which tabulate the total value for rubber production respectively. The assumed ± 5% variations in the data from the actual data are denoted by $\underline{a}$ and $\overline{a}$ respectively. $\Pr_i$ is the probability assigned to each data, and observed as the proportional production of different mills distributed in the states of Malaysia; shows the randomness.

Table 1. Dataset of rubber production

	Rubber production		
Year	Dry	Latex	Total production
2000	774,248.00	153,360	927,608
2001	761,594.00	120,473	882,067
2002	775,334.00	114,498	889,832
2003	854,619.00	131,028	985,647
2004	960,841.00	207,894	1,168,735
2005	935,529.00	190,494	1,126,023
2006	1,073,698.00	209,934	1,283,632
2007	1,023,190.00	176,363	1,199,553
2008	918,656.00	153,709	1,072,365
2009	746,106.00	110,913	857,019
2010	846,813.00	92,428	939,241
2011	916,270.00	79,940	996,210
2012	846,813.00	75,985	922,798

Table 2. Data Preparation for dry production

$\underline{\theta P}_i$	Pr_i	θP_i	Pr_i	$\overline{\theta P}_i$	Pr_i
774,247.95	0.33	774,248	0.33	774,248.05	0.33
761,593.95	0.33	761,594	0.33	761,594.05	0.33
775,333.95	0.33	775,334	0.33	775,334.05	0.33
854,618.95	0.33	854,619	0.33	854,619.05	0.33
960,840.95	0.33	960,841	0.33	960,841.05	0.33
935,528.95	0.33	935,529	0.33	935,529.05	0.33
1,073,697.95	0.33	1,073,698	0.33	1,073,698.05	0.33
1,023,189.95	0.33	1,023,190	0.33	1,023,190.05	0.33
918,655.95	0.33	918,656	0.33	918,656.05	0.33
746,105.95	0.33	746,106	0.33	746,106.05	0.33
846,812.95	0.33	846,813	0.33	846,813.05	0.33
916,269.95	0.33	916,270	0.33	916,270.05	0.33
846,812.95	0.33	846,813	0.33	846,813.05	0.33

4.2 Estimating the Coefficient

This section is spent to demonstrate the linear programming for rubber production in which the coefficient is estimated. The model is developed based on the algorithm for fuzzy random regression as explained in the section 3. The confidence interval which is necessary to treat fuzzy random uncertainties is calculated beforehand.

Regression Model for Rubber production

Equation (4) is the partial linear program for rubber production.

Rubber Production Model:

$$
\begin{array}{ll}
\min_{\bar{A}_{production}} & J(\bar{A}_{production}) = \sum_{k=1}^{2} (\bar{A}_k^r - \bar{A}_k^l) \\
\text{subject to:} & \bar{A}_1^r \geq \bar{A}_1^l \geq 0, \\
& \bar{A}_2^r \geq \bar{A}_2^l \geq 0, \\
& (7.7042\times10^5)\bar{A}_1^l + (1.5260\times10^5)\bar{A}_2^l \leq 9.2302\times10^5, \\
& (7.5782\times10^5)\bar{A}_1^l + (1.1988\times10^5)\bar{A}_2^l \leq 8.7770\times10^5, \\
& (7.7150\times10^5)\bar{A}_1^l + (1.1393\times10^5)\bar{A}_2^l \leq 8.8543\times10^5, \\
& (8.5039\times10^5)\bar{A}_1^l + (1.3038\times10^5)\bar{A}_2^l \leq 9.8077\times10^5, \\
& ^{5}, \\
& (9.2081\times10^5)\bar{A}_1^r + (8.0335\times10^4)\bar{A}_2^r \geq 10.0114\times10^5, \\
& (8.5100\times10^5)\bar{A}_1^r + (7.6361\times10^4)\bar{A}_2^r \geq 9.2737\times10^5,
\end{array}
\tag{4}
$$

The regression model (4) was applied to the datasets and were performed using Lingo@ computer software to compute the coefficient values.

5 Results and Discussion

This section explains the result obtained from fuzzy random regression of problem in Equation (4). The linear program is solved and the comparison was made by the result with those obtained by the fuzzy regression approach.

5.1 Coefficient Result

The regression models (4) were developed based on computation of confidence interval for rubber production data. Table 4 shows the coefficient obtained for rubber. The coefficient result for the rubber production as tabulated in Table 3 shows the values estimated from fuzzy random regression and fuzzy regression. The evaluation of attributes x_1 and x_2 indicate the x_2 is significant to the total evaluation due to its higher coefficient. Meanwhile, the evaluation of parameters for x_1 and x_2 which are planted area of (1.00, 0.99), companies of (1.00, 0.99), employment (0.34, 1.99) and consumption of (1.00, 0.99) show the significant of x_1 to the total evaluation. In general, both fuzzy regression and fuzzy random regression is capable of dealing such data to estimate the coefficient values. The decision results in the form of coefficient and its width of decision factor, x_i .The fuzzy random regression model had a wider coefficient width because of the consideration of the confidence interval in its evaluation. The width in this evaluation plays an important role, as it reflects natural human judgment. A wider width indicates that the evaluation can captures more information under fuzzy judgments.

Table 3. Coefficient result for the rubber production

Item	Attributes	FRRM		FRM	
		coefficient	width	coefficient	width
Production	x_1	0.99	0.00	0.99	0.00
	x_2	1.00	0.00	1.00	0.00
Planted area	x_1	1.00	0.00	0.99	0.00
	x_2	0.99	0.00	0.99	0.00
Employment	x_1	0.34	0.00	1.00	0.00
	x_2	1.99	0.00	0.99	0.00
Companies	x_1	1.00	0.00	1.00	0.00
	x_2	0.99	0.00	0.99	0.00
Consumption	x_1	1.00	0.00	0.99	0.00
	x_2	0.99	0.00	1.00	0.00

*FRRM – Fuzzy Random Regression Model
*FRM – Fuzzy Regression Model

5.2 Development of Linear Model

The fuzzy random regression model with confidence interval for the rubber data were then defined as follow:

$$\begin{aligned}\bar{Y}_{production} &= \left(\bar{A}_i^{L,R}\right)_T I\left[e_{X_i},\sigma_{X_i}\right] \\ &= (0.99, \quad 0.99)_T I\left[e_{X_1},\sigma_{X_1}\right] + (1.00, \quad 1.00)_T I\left[e_{X_2},\sigma_{X_2}\right]\end{aligned} \tag{5}$$

$$\begin{aligned}\bar{L}_{plantedarea} &= \left(\bar{A}_i^{L,R}\right)_T I\left[e_{X_i},\sigma_{X_i}\right] \\ &= (1.00,1.00)_T I\left[e_{X_1},\sigma_{X_1}\right] + (0.99, \quad 0.99)_T I\left[e_{X_2},\sigma_{X_2}\right]\end{aligned} \tag{6}$$

$$\begin{aligned}\bar{N}_{employment} &= \left(\bar{A}_i^{L,R}\right)_T I\left[e_{X_i},\sigma_{X_i}\right] \\ &= (0.34, \quad 0.34)_T I\left[e_{X_1},\sigma_{X_1}\right] + (1.99, \quad 1.99)_T I\left[e_{X_2},\sigma_{X_2}\right]\end{aligned} \tag{7}$$

$$\begin{aligned}\bar{C}_{companies} &= \left(\bar{A}_i^{L,R}\right)_T I\left[e_{X_i},\sigma_{X_i}\right] \\ &= (1.00, \quad 1.00)_T I\left[e_{X_1},\sigma_{X_1}\right] + (0.99, \quad 0.99)_T I\left[e_{X_2},\sigma_{X_2}\right]\end{aligned} \tag{8}$$

$$\begin{aligned}\bar{D}_{consumption} &= \left(\bar{A}_i^{L,R}\right)_T I\left[e_{X_i},\sigma_{X_i}\right] \\ &= (1.00, \quad 1.00)_T I\left[e_{X_1},\sigma_{X_1}\right] + (0.99, \quad 0.99)_T I\left[e_{X_2},\sigma_{X_2}\right]\end{aligned} \tag{9}$$

In statistical analysis, regression is widely been used for prediction and forecasting. In this study, the regression analysis is used to estimate the coefficient under fuzzy random situation where this coefficient can be used to predict the future production in the practical of rubber industry. The model results under predict by 8795.162. As for the mean absolute error, the model yield 8795.162. The model yields 8850.691 as the RMSE. A residual plot is shown by scatter plotting where the x-axis is the predicted value x and the y-axis is the residual of x. The residual plot as shown in Fig. 1, the model provides a random scattering of points.

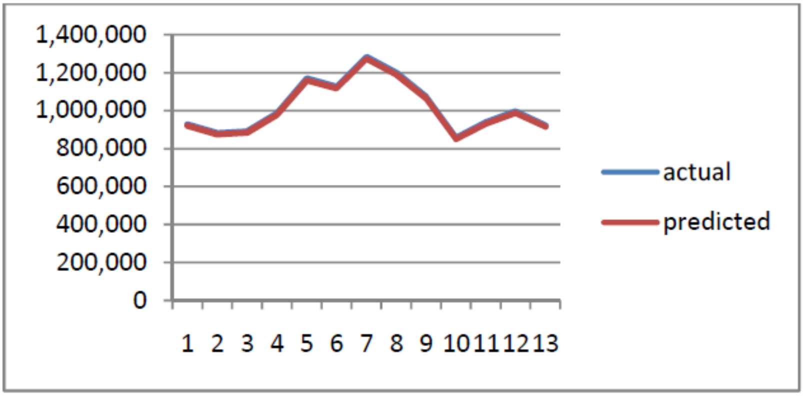


Fig. 1. Actual and Predicted Production

6 Conclusions

This analysis permits the estimation of the coefficient values using historical data for cases in which the data simultaneously contain fuzziness and randomness. The previous patterns of outcomes are included into the future prediction or decision. The results demonstrate that the proposed method of using fuzzy random regression can determine the important coefficient values and the uncertainty on the data are treated. We demonstrate that the fuzzy random regression enables to reduce the difficulty of determining the coefficient values for developing production planning model. The work described in this study reveals that fuzzy random regression can be used to better facilitate the decision making process, specifically to extract important coefficients from among decision factor and to address simultaneous fuzzy random information.

Acknowledgments. The authors express her appreciation to the University Tun Hussein Onn Malaysia (UTHM) and Research Acculturation Grant Scheme (RAGS) Vot R044, and, to the Malaysian Rubber Board Statistic for providing research data and discussion.

References

1. Smith, T.F., Waterman, M.S.: Identification of Common Molecular Subsequences. J. Mol. Biol. 147, 195–197 (1981)
2. May, P., Ehrlich, H.C., Steinke, T.: ZIB Structure Prediction Pipeline: Composing a Complex Biological Workflow through Web Services. In: Nagel, W.E., Walter, W.V., Lehner, W. (eds.) Euro-Par 2006. LNCS, vol. 4128, pp. 1148–1158. Springer, Heidelberg (2006)
3. Foster, I., Kesselman, C.: The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann, San Francisco (1999)
4. Czajkowski, K., Fitzgerald, S., Foster, I., Kesselman, C.: Grid Information Services for Distributed Resource Sharing. In: 10th IEEE International Symposium on High Performance Distributed Computing, pp. 181–184. IEEE Press, New York (2001)
5. Foster, I., Kesselman, C., Nick, J., Tuecke, S.: The Physiology of the Grid: an Open Grid Services Architecture for Distributed Systems Integration. Technical report, Global Grid Forum (2002)
6. National Center for Biotechnology Information, http://www.ncbi.nlm.nih.gov, Montgomery, D.C., Peck, E.A., Vining, G.G.: Introduction to linear regression analysis, vol. 821. Wiley (2012)
7. Griffiths, T.L., Tenenbaum, J.B.: Predicting the future as Bayesian inference: People combine priorknowledge with observations when estimating duration and extent 140(4), 725–743 (2011)
8. Cave, W.C.: Prediction Theory for Control System (2011)
9. Yang, M.S., Ko, C.H.: On cluster-wise fuzzy regression analysis. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 27(1), 1–13 (1997)

10. González-Rodríguez, G., Blanco, Á., Colubi, A., Lubiano, M.A.: Estimation of a simple linear regression model for fuzzy random variables. Fuzzy Sets and Systems 160(3), 357–370 (2009)
11. Watada, J.: Building models based on environment with hybrid uncertainty. In: 2011 4th International Conference on Modeling, Simulation and Applied Optimization (ICMSAO), pp. 1–10. IEEE (April 2011)
12. Nureize, A., Watada, J.: Multi-level multi-objective decision problem through fuzzy random regression based objective function. In: 2011 IEEE International Conference on Fuzzy Systems (FUZZ), pp. 557–563. IEEE (June 2011)
13. Watada, J., Wang, S., Pedrycz, W.: Building confidence-interval-based fuzzy random regression models. IEEE Transactions on Fuzzy Systems 17(6), 1273–1283 (2009)
14. Näther, W.: Regression with fuzzy random data. Computational Statistics & Data Analysis 51(1), 235–252 (2006)
15. Kwakernaak: Fuzzy random variables—I. Definitions and Theorems. Information Sciences 15(1), 1–29 (1978)
16. Market Watch 2012. The Rubber Sector in Malaysia, `http://www.malaysia.ahk.de` (retrieved October 22, 2013)
17. Malaysian Investment Development Authority. Rubber-based industry, `http://www.mida.gov.my` (retrieved on October 10, 2013)
18. Malaysian Rubber Export Promotion Council, `http://www.mrepc.gov.my` (retrieved on October 10, 2013)
19. Malaysian Rubber Board.Natural Rubber Statistic, `http://www.lgm.gov.my` (retrieved September 1, 2013)

Honey Bees Inspired Learning Algorithm: Nature Intelligence Can Predict Natural Disaster

Habib Shah, Rozaida Ghazali, and Yana Mazwin Mohmad Hassim

Faculty of Computer Science and Information Technology
Universiti Tun Hussein Onn Malaysia (UTHM)
Parit Raja, 86400 Batu Pahat, Johor, Malaysia
habibshah.uthm@gmail.com, {rozaida,yana}@uthm.edu.my

Abstract. Artificial bee colony (ABC) algorithm which used the honey bee intelligence behaviors, is a new learning technique comparatively attractive for solving optimization problems. Artificial Neural Network (ANN) trained with the ABC algorithm normally has poor exploration and exploitation processes due to the random and similar strategies for finding best position of foods. Global artificial bee colony (Global ABC) and Guided artificial bee colony (Guided ABC) algorithms used to produce enough exploitation and exploration strategies respectively. Here, a hybrid of Global ABC and Guided ABC is proposed called Global Guided ABC (GG-ABC) algorithm, for getting balance and robust exploitation and exploration process. The experimental result shows that the GG-ABC performed better than other algorithms for prediction of earthquake hazards.

Keywords: Guided Artificial Bee Colony, Global Artificial Bee Colony, Global Guided Artificial Bee Colony algorithm, Earthquake prediction.

1 Introduction

In the past decade, the fracture of earth, flow of rocks, movements of tectonic plates, heat waves temperature and the high range of sea waves has been focused by geologists and engineers for saving human life as well as economic of the region. These sources may be the most important rule in earthquake, water level height and tsunami occurrence called seismic signals or natural hazards. Seismic events, especially earthquake is the most costly natural hazards faced by the nation in which they occur without prior warning and may cause severe injuries [1]. The intensity of occurrence of such event creates disasters and can change human and animals lives [2].

In recent years much has been learned from natural disasters and risk to infrastructure systems. In 2011, natural disaster costs (US$ 365.6 billion) were the highest of the decade, accounting for almost 1.5 times the direct losses reported in 2005 (US$ 248 billion, 2011 prices) [3]. The countless lives in earthquake risk areas can be saved, and the human and economic losses caused by these events can be reduced [3].

Seismic time series signals prediction are the challenges for researchers which can be solved through proper ANN model and robust learning algorithms [4]. The robust

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_21, © Springer International Publishing Switzerland 2014

learning algorithm has the properties like high prediction accuracy and very less error. This can be achieved through optimal weight values, accepted NN model, getting the behaviours of data, careful selection of learning parameters, data pre-processing methods, suitable number of layers, nodes and appropriate activation function [5].Traditional learning algorithms of ANN like Backpropagation (BP) is well-known for getting the solution in different applications with various tasks; however they easily got struck in local minima.

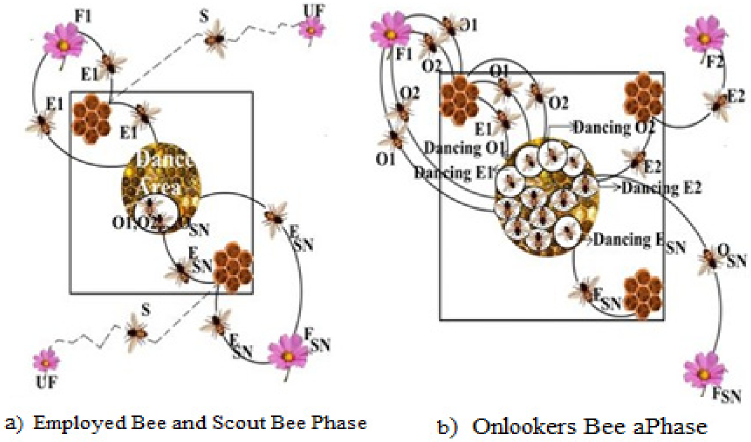
The swarm intelligence (SI) algorithms used to train ANN for finding best parameter values. Besides solving other optimization problems through ABC, getting the optimal weight values for ANN through training procedures based artificial behaviours of bees getting best results made it more famous. All types of SI algorithms are successfully applied to statistical and engineering tasks, and showed the best performance from others local search algorithms [6]. SI learning algorithms with optimal exploration and exploitation procedures provide high performance for a given task.

In this work, the GG-ABC algorithm is proposed for getting high accuracy for natural disaster prediction, like earthquake magnitude using natural inspiration techniques of honey bees. The performance of GG-ABC is compared with standard ABC, Guided ABC and Global ABC algorithms. The GG-ABC algorithm is used successfully to train Multilayer Perceptron (MLP) for earthquake magnitude prediction.

2 Honey Bees Inspired Learning Algorithms

Bio inspired intelligence learning techniques deals with natural and artificial systems that composed of many individuals agents, with coordinated patterns using decentralized control, adaptively, unity, cooperation and self organization properties. These are evolutionary algorithms (EAs), particle swarm optimization (PSO), ant colony (ACO), cuckoo search (CS), artificial bee colony (ABC) and so on, can be used for solving optimization problems [7]. Bio inspired techniques focuses on the collective behaviours that result from the local interactions of the individuals with each other and with their environment, through the artificial gathering of ants, fish, birds, herds of land animals, and honey bees. These algorithms are relatively new optimization techniques which have been shown to be competitive to other non-swarm algorithms. These natural intelligence techniques are popular for exploration and exploitation process, where robust swarms agents provide strong amount exploration and exploitation with balance quantity [8].

The bees intelligence algorithms well-known with their nature beauty, robust performance, global and neighbour acceptable knowledge. Social insect colonies can be considered as dynamic system of gathering information from the environment and adjusting its behaviour in accordance to it. Generally, all social insect colonies behave according to their own division of labours related to their morphology. Bee system, consists of two essential components. These components are foods and foragers, where the amount of food source depends on various parameters such as its closeness to the nest, richness of energy and ease of extracting this energy [9]. There are three types bees working as a team namely: employed and onlooker are exploiters, while scout function is to explore. Fig 1 shows the behavior of honey bee foraging for nectar.



a) Employed Bee and Scout Bee Phase b) Onlookers Bee aPhase

Fig. 1. Minimal Model of Foraging Behaviour of Honey Bees

where; UF: Unemployed Foragers, S: Scout, EF1:Employed Forager, EF2: Employed Forager, U: Undiscovered Food Sources R: Recruited Forager, SN: Food Sources. Mathematically, collecting exploration and exploitation respectively through standard ABC can be written as:

$$v_{ij} = x_{ij} + \phi_{ij}\left(x_{ij} - x_{kj}\right) \tag{1}$$

$$x_{ij}^{rand} = x_{ij}^{min} + rand\left(0,1\right)\left(x_{ij}^{max} - x_{ij}^{min}\right) \tag{2}$$

where, k is a solution in the neighborhood of i, φ is a random number. The standard ABC has been extended by researchers to many different versions such as gbest, guided best, so best so far and hybridization with local for various tasks [8].

2.1 Global Artificial Bee Colony (Global ABC) Algorithm

The most favorable solution for optimization problems are through exploration and exploitation which should be balanced with effective quantity for each individual agents of swarm based learning algorithms like ABC and PSO. To increase exploitation step within specified area, Global ABC algorithm [10] which is an optimization tool provides stochastic search procedure in which agents are adapted by the global best artificial bees with time, and the bee's aim is to discover the best places of global food sources [7]. This gbest strategy has good exploitation amount in the employed and onlooker bees sections. Unfortunately, the Global ABC cannot provide the strong exploration, because this technique guides the employed and onlookers agents only, so that exploration procedure cannot be improved. The gbest search strategy used in employed and onlooker are given in Eq (3):

$$v_{ij} = x_{ij} + \phi_{ij}\left(x_{ij} - x_{kj}\right) + c_1 rand\left(0,1\right)\left(x_j^{best} - x_{ij}\right) + c_2 rand\left(0,1\right)\left(y_j^{best} - x_{ij}\right) \tag{3}$$

where y shows best food source, C_1 and C_2 are two positive constant values, $x^{best}{}_J$ is the j_{th} element of the global best solution found so far, $y^{best}{}_J$ is the j_{th} element of the best solution in the current iteration.

2.2 Guided Artificial Bee Colony (Guided ABC) Algorithm

The Guided ABC algorithm is an advanced version of standard ABC, which proposed for improving the exploration procedure through scout bee searching strategy [11]. Guided ABC has heuristic procedure for the solution of combinatorial optimizations and discrete problems that has inspired by honey bees. The Guided ABC is an attractive algorithm because it use the real bee agent for solving the optimization. In the Guided ABC algorithm the scout bee has been guided to get optimal food source position instead of random methods, which used to increase the exploration process for given task. The scout will generate a new solution through global best knowledge information. The global best experience will modify with the following best guided strategy as:

$$v_{ij} = x_{ij} + \phi * (x_{ij} - x_{kj}) + (1-\phi) * (x_{ij} - x_{best\,j}) \tag{4}$$

The Guided ABC will increase the capabilities of the standard ABC to produce new best solutions located near the feasible area. This technique is successfully used for constrained optimization problems with enough exploration [11], however the employed and onlookers bees have random strategy with poor exploitation.

3 The Proposed Global Guided Artificial Bee Colony Algorithm

Natural intelligence algorithms become attractive due to their robust searching ability through the neighbour agent information. Habitually the exploration and exploitation are the strategies which can be improved with the best movement of neighbour agent information in standard bees algorithm. The performance of ABC algorithm depends on agent dancing and strong power of intelligence, so if the agent has enough intelligence, it can provide strong exploration and exploitation process in the particular area for given problems. In bee algorithm, the employed and onlookers bees have the duty of exploitation procedures. While, the scout bees are used for getting enough amount of exploration. The Global ABC has successfully improved the exploitation through global best bees methods. Furthermore the guided ABC has outstanding performance with strong exploration due to the guided scout bees.

Combining the gbest agent strategies of Global ABC and Guided ABC algorithms for getting enough amount of exploration and exploitation with balance quantity. The new hybrid algorithm called Global Guided Artificial Bee Colony (GG-ABC) algorithm. The GG-ABC agents used global employed / onlookers and guided scout bees are used to find the best food source. The GG-ABC will merge their best finding approaches with original ABC by the following steps. The exploitation procedure will increase through Eq 5 using global ABC strategy, and exploration with Eq 6, through guided scout bees. The pseudo code of the proposed GG-ABC algorithm detailed as:

```
Initialize the food source positions
REPEAT
  Global Employed Bee Stage
  Global Onlooker Bee Stage
  Guided Scout Bee Stage
  Memorize the best solution achieved until now
Until Maximum Cycle Number (MCN).
```

Mathematically the exploitation and exploration process can be improved and balance with enough amount through the following equations.

Global Employed / Onlooker Bee Phase

$$v_{ij} = x_{ij} + \phi_{ij}\left(x_{ij} - x_{kj}\right) + c_1 rand(0,1)\left(x_j^{best} - x_{ij}\right) + c_2 rand(0,1)\left(y_j^{best} - x_{ij}\right) \tag{5}$$

Guided Scout Bee Phase

$$v_{ij} = x_{ij} + \phi * \left(x_{ij} - x_{kj}\right) + (1 - \phi) * (x_{ij} - x_{best\ j}) \tag{6}$$

Where v_{ij} shows best food source, c_1 and c_2 are two constant values which is 2.5 and 1.5 for this study respectively, $\mathrm{x}_{\mathrm{best\ j}}$ is the j_{th} element of the global best solution found so far, $y_{\mathrm{j}}^{\mathrm{best}}$ is the j_{th} element of the best solution in the current iteration, ϕ_{ij} is a uniformly distributed real random number in the range [-1, 1].

4 Experimental Design

In this research, the natural disaster earthquake significant parameter called magnitude measured through the Richter scale is used for prediction. The univariate time series data was taken from the Southern California Earthquake Data Centre (SCEDC) holdings for November and December 2013 (SCEDC, 2013) and Northern California Earthquake Data Centre (NCEDC) for November and December of 2012. Seismic time series data are highly non-linear and non-stationary signals which exhibit high volatility, complexity and noise. Therefore, seismic time series need preprocessing before presenting them to MLP for training and testing.

In this work, GG-ABC algorithm is used to train MLP for the South and Northern California earthquake time series data for prediction tasks. To calculate the performance of the ABC, Global ABC, Guided ABC and GG-ABC algorithms by Mean of Mean Square Error (MSE), Normalized Mean Square Error (NMSE) and Signal to Noise Ratio (SNR). The stopping criteria for ABC and Global ABC, Guided ABC and GG-ABC stopped on 2000 MCN with 20 colony size.

5 Simulation Results and Analysis

The proposed GG-ABC algorithm was utilized to predict the occurrence of earthquake magnitude value within seconds of South and Northern California earthquake dataset. These time series were fed to the MLP to capture the underlying rules of the earth movement in the specified regions of South and Northern California. For comparison, the evaluation of each learning algorithm used for the prediction tasks are summarized based on average results of 10 runs, which are further explained in the following subsections.

5.1 South California Earthquake Prediction

The best average evaluation results using all learning algorithms for South California earthquake prediction are given in Tables 1 to 3 and Figures 2 to 4 respectively. In Table 1, the MSE for testing data set is presented. The proposed GG-ABC gives small error from other algorithms in terms of MSE and NMSE for the South California earthquake prediction task as given in Table 1 and 2.

Table 1. Average MSE on out of sample data for South California earthquake prediction

NN Structure	ABC	Global ABC	Guided ABC	GG-ABC
5-2-1	0.000723	0.0004912	0.0005326	**0.0001326**
5-3-1	0.000521	0.0004201	0.0003301	**0.0001021**
5-5-1	0.000501	0.0003101	0.0003110	**0.0001007**
5-7-1	0.000442	0.0002981	0.0002089	**0.0000531**
5-9-1	0.000431	0.0001781	0.0001162	**0.0000330**

Meanwhile the maximum SNR values for out of sample data from average simulation results are given in Table 3. The maximum SNR on unseen data reached to 36.9521 value by the proposed GG-ABC, which is better than other learning algorithms for South California earthquake prediction. Throughout the training and testing process of GG-ABC algorithm.

Table 2. Average NMSE on out of sample data for South California earthquake prediction

NN Structure	ABC	Global ABC	Guided ABC	GG-ABC
5-2-1	0.161091	0.141021	0.246391	**0.116301**
5-3-1	0.111025	0.101530	0.206952	**0.099844**
5-5-1	0.101034	0.101191	0.124639	**0.054652**
5-7-1	0.089511	0.094154	0.102411	**0.022942**
5-9-1	0.076101	0.052101	0.028345	**0.010139**

Table 3. Average SNR on out of sample data for South California earthquake prediction

NN Structure	ABC	Global ABC	Guided ABC	GG-ABC
5-2-1	24.4815	26.0012	25.0311	**29.2982**
5-3-1	24.0012	27.0234	26.0209	**30.8873**
5-5-1	25.1093	28.1981	28.1961	**33.1209**
5-7-1	25.0061	28.0283	29.0103	**32.1892**
5-9-1	26.4211	30.4299	30.0911	**36.9521**

Fig. 2 represents learning curves for the proposed GG-ABC algorithms applied to a MLP. For each topology, the network was trained on successive input examples for the number of iterations shown. From Fig 2, which is the convergence curve of proposed GG-ABC algorithm for South California earthquake magnitude prediction shows fast convergence on 200 Maximum Cycle Number (MCN). This is because of strong exploration and exploitation procedures through global and guided technique. The prediction results the out-of-sample data are reported in Fig 3 and 4. After completing several simulations for predicting earthquake magnitude based on the past historical data using GG-ABC algorithm, it is concluded that the average error for prediction of earthquake is smaller. For the presentation purpose, the first 100 samples of data has been selected from earthquake magnitude prediction. From Fig 3 and 4 show that the predicted earthquake signal's are close to the real values.

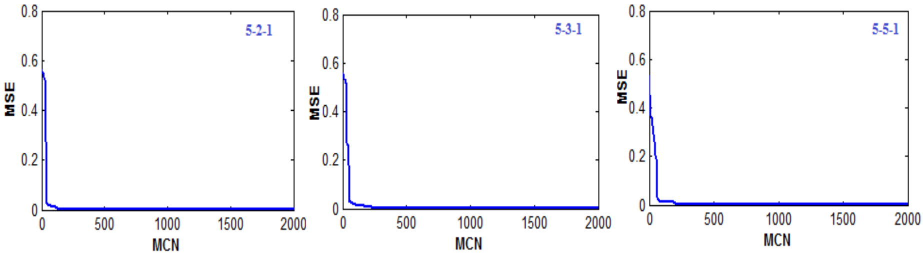


Fig. 2. Learning curves of South California earthquake using proposed GG-ABC algorithm

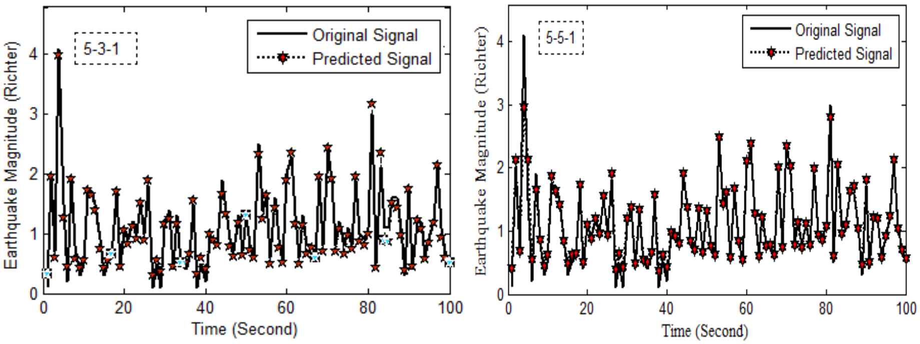


Fig. 3. Prediction of South California earthquake on out of sample data by GG-ABC algorithm

5.2 Northern California Earthquake Prediction

The best average simulation results using all algorithms for Northern California earthquake prediction are given in the Tables 4 to 6 and Fig 5, 6 and 7. In Table 4, the MSE for testing data set is presented. The results for earthquake prediction from Tables 4 to 6 obviously demonstrated that the GG-ABC algorithm in all cases achieved very small error compared to other algorithms. Table 6 clearly shows that the highest SNR values were obtained by the proposed GG-ABC algorithm.

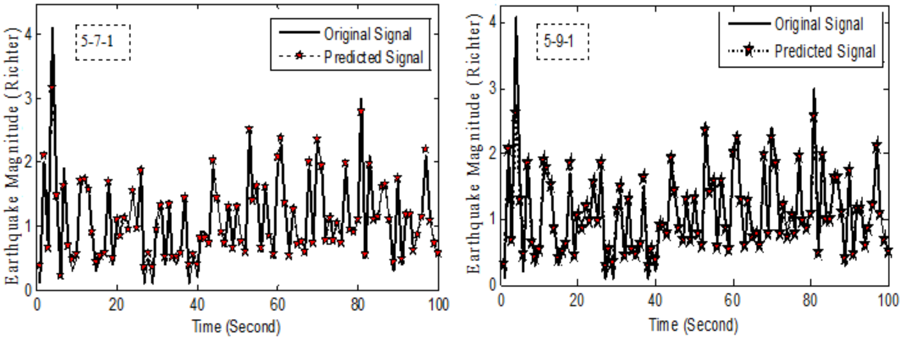


Fig. 4. Prediction of South California earthquake on out of sample data by GG-ABC algorithm

Table 4. Average MSE on out of sample data for Northern California earthquake prediction

NN Structure	ABC	Global ABC	Guided ABC	GG-ABC
5-2-1	0.000321	0.000273	0.000365	**0.0001383**
5-3-1	0.000392	0.000211	0.000391	**0.0003092**
5-5-1	0.000290	0.000202	0.000253	**0.0001857**
5-7-1	0.000359	0.000120	0.000223	**0.0001002**
5-9-1	0.000220	0.000147	0.000112	**0.0000992**

Table 5. Average NMSE on out of sample data for Northern California earthquake prediction

NN Structure	ABC	Global ABC	Guided ABC	GG-ABC
5-2-1	0.178905	0.110931	0.216598	**0.110973**
5-3-1	0.128921	0.102078	0.127091	**0.098752**
5-5-1	0.105611	0.097523	0.101102	**0.025691**
5-7-1	0.079013	0.070923	0.092056	**0.029786**
5-9-1	0.049625	0.042987	0.068732	**0.011023**

Table 7 contains the maximum SNR for Northern California earthquake magnitude prediction with average simulation results using the above learning algorithms. The maximum SNR for out of sample data obtained by Global ABC reached to 30.4986, which is less from other algorithms. The GG-ABC reached to 38.1218 value for SNR with Nine hidden nodes, which is better than all learning algorithms. So the proposed GG-ABC algorithm has the overall outstanding maximum SNR value.

Table 6. Average SNR on out of sample data for Northern California earthquake prediction

NN Structure	ABC	Global ABC	Guided ABC	GG-ABC
5-2-1	28.1201	29.1832	28.4572	**34.2236**
5-3-1	29.9826	30.0714	30.1051	**35.8875**
5-5-1	30.9542	30.1561	30.1002	**36.4389**
5-7-1	30.4321	32.0139	31.9823	**36.1702**
5-9-1	30.4986	33.4216	31.5627	**38.1218**

The Table 6 shows that the performance of GG-ABC is better when compared with other learning algorithms. The proposed algorithm played quite an important role in a network's performance. Fig. 5, obtained by GG-ABC algorithm in training step for earthquake seismic time-series, where the MSE is also stable and converged quickly.

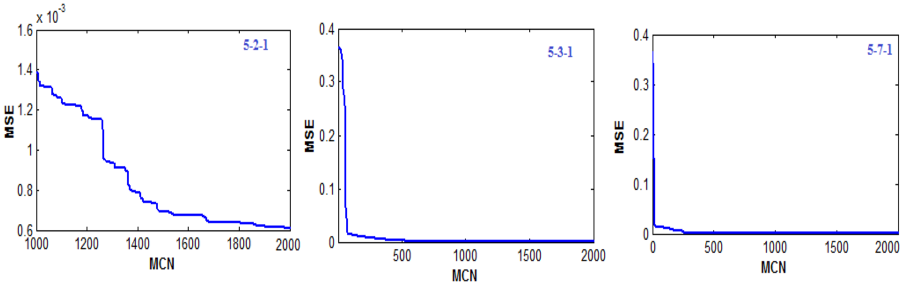


Fig. 5. Learning curves of Northern California earthquake using proposed GG-ABC algorithm

The prediction of earthquake time-series for unseen data using GG-ABC is given in Fig. 6 and 7 with a different number of hidden nodes. From these figures, the best average prediction results through GG-ABC indicate close to actual signal.

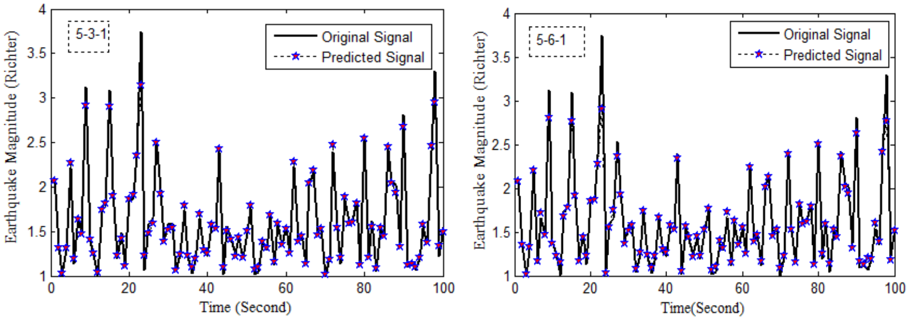


Fig. 6. Northern California earthquake prediction on out of sample data by GG-ABC algorithm

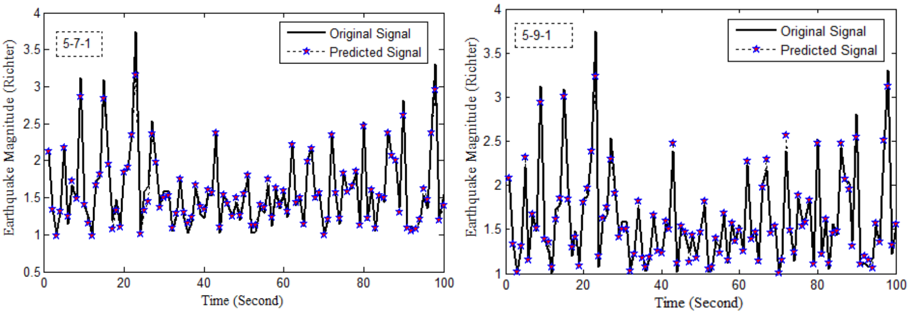


Fig. 7. Northern California earthquake prediction on out of sample data by GG-ABC algorithm

Fig. 6 and 7 shows the prediction performance of GG-ABC compared with the desired values as graphed for Northerly California earthquake prediction. The above simulation results demonstrate, that GG-ABC algorithm has successfully predicted the earthquake magnitude of South and Northern California with less error. The non-linear dynamical behavior is induced by the bee nature intelligence. Therefore, it leads to the best input output mapping and an optimum prediction.

6 Conclusion

In this paper, GG-ABC algorithm is used to find the optimal weight set for MLP through bees. It has been proved that the GG-ABC algorithm is successfully used in improving bees intelligence which increase the exploration and exploitation process. The results show that the GG-ABC algorithm possesses high performance in earthquake prediction than other algorithms. Hence, the GG-ABC algorithm may be a good alternative to deal with earthquake prediction. It is interesting that despite the high evolution in technology, simply nature can be used to predict natural disasters using an artificial intelligence.

Acknowledgment. The authors would like to thank University Tun Hussein Onn Malaysia and Ministry of High Education (MOHE) for supporting this research under the ERGS Vote 0882.

References

1. Alves, E.I.: Earthquake Forecasting Using Neural Networks: Results and Future Work. Nonlinear Dynamics 44(1-4), 341–349 (2006)
2. Shah, H., et al.: Global Artificial Bee Colony-Levenberq-Marquardt (GABC-LM) Algorithm for Classification. International Journal of Applied Evolutionary Computation (IJAEC) 4(3), 58–74 (2013)
3. Zette, R. (ed.): Ifrcrcs, World Disaster Report, in Focus on forced migration and displacement. International Federation of Red Cross and Red Crescent Societies:17, Chemin des Crêts, P.O.Box 372 CH-1211 Geneva 19, Switzerland, p. 310 (2012)
4. Adeli, H., Panakkat, A.: A probabilistic neural network for earthquake magnitude prediction. Neural Networks 22(7), 1018–1024 (2009)
5. Kasabov, N.K.: Functionally reconfigurable general purpose parallel machines and some image processing and pattern recognition applications. Pattern Recognition Letters 3(3), 215–223 (1985)
6. Karaboga, D., Akay, B., Ozturk, C.: Artificial Bee Colony (ABC) Optimization Algorithm for Training Feed-Forward Neural Networks. In: Torra, V., Narukawa, Y., Yoshida, Y. (eds.) MDAI 2007. LNCS (LNAI), vol. 4617, pp. 318–329. Springer, Heidelberg (2007)
7. Zhu, G., Kwong, S.: Gbest-guided artificial bee colony algorithm for numerical function optimization. Applied Mathematics and Computation 217(7), 3166–3173 (2010)
8. Shah, H., Ghazali, R., Nawi, N.M.: Global Artificial Bee Colony Algorithm for Boolean Function Classification. In: Selamat, A., Nguyen, N.T., Haron, H. (eds.) ACIIDS 2013, Part I. LNCS, vol. 7802, pp. 12–20. Springer, Heidelberg (2013)

9. Adil, B., Lale, Ö., Pınar, T. (eds.): Artificial Bee Colony Algorithm and Its Application to Generalized Assignment Problem, Turkey (2007)
10. Peng, G., Wenming, C., Jian, L.: Global artificial bee colony search algorithm for numerical function optimization. In: 2011 Seventh International Conference on Natural Computation (ICNC) (2011)
11. Tuba, M., Bacanin, N., Stanarevic, N.: Guided artificial bee colony algorithm. In: Proceedings of the 5th European Conference on European Computing Conference, pp. 398–403. World Scientific and Engineering Academy and Society (WSEAS), Paris (2011)

Hybrid Radial Basis Function with Particle Swarm Optimisation Algorithm for Time Series Prediction Problems

Ali Hassan and Salwani Abdullah

Center for Artificial Intelligence Technology,
Faculty of Information Science and Technology,
Universiti Kebangsaan Malaysia (UKM)
aa87hassan@yahoo.com,
salwani@ftsm.ukm.my

Abstract. Time Series Prediction (TSP) is to estimate some future value based on current and past data samples. Researches indicated that most of models applied on TSP suffer from a number of shortcomings such as easily trapped into a local optimum, premature convergence, and high computation complexity. In order to tackle these shortcomings, this research proposes a method which is Radial Base Function hybrid with Particle Swarm Optimization algorithm (RBF-PSO). The method is applied on two well-known benchmarks dataset Mackey-Glass Time Series (MGTS) and Competition on Artificial Time Series (CATS) and one real world dataset called the Rainfall dataset. The results revealed that the RBF-PSO yields competitive results in comparison with other methods tested on the same datasets, if not the best for MGTS case. The results also demonstrate that the proposed method is able to produce good prediction accuracy when tested on real world rainfall dataset as well.

Keywords: Time series prediction, Radial Base Function, Particle Swarm Optimization, Mackey-Glass Time Series, Competition on Artificial Time Series.

1 Introduction

Time Series Prediction (TSP) is the outcome of a phenomenon over time in the form of a time series, and it is normally carried out by investigating patterns in historical data and by speculating on how a future trend will behave according to its past pattern. Many efforts have been made over the past several decades to develop and improve TSP models. TSP has been used in a variety of complex systems and applications such as financial market prediction, weather and environmental state prediction, electric utility load forecasting, reliability forecasting and so on, which help in decision making and planning.

Many researchers, such as [27], [1], [5], [17], introduced modern information analysis techniques for linear and nonlinear systems. Despite the amount of models

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_22, © Springer International Publishing Switzerland 2014

that have been introduced, there is still a gap to question their predicting abilities. Most of the models suffer from a number of shortcomings such as easily falling into local minima [11], dependence between the quality of the results and the specific characteristics of the time series data [1], slow convergence speed or premature convergence [11], i.e. the model converges to the local optimum before it has time to explore another part of the fitness landscape containing the global optimum, time complexity [26], and determining the tuning parameter appropriately [19].

The abovementioned shortcomings of previous models are the main motivating factors that led to the proposed hybridization of the Radial Basis Function with the Particle Swarm Optimization algorithm (RBF-PSO) which is aimed at overcoming the challenges of time series prediction.

This paper is organized as follows: Section 2 outlines our proposed method RBF-PSO. Experiments and Discussion are presented in Section 3. Finally, the last section summarizes this study.

2 Proposed Method

2.1 Radial Basis Function

Radial basis functions (RBFs) are embedded into a two-layer, feed-forward neural network. Such a network is characterized by a set of inputs and a set of outputs. In between the inputs and the outputs there is a layer of processing units called hidden layers. Each of them implements a radial basis function. The RBF network is a network where the activation of the hidden layer is based on the distance between the input vector and the prototype vector as shown in Eqn 1. Various functions have been tested as activation functions for RBF [4]. In time series modeling, the preferred activation function is the Gaussion function [3], [12], [14], [18]. The Gaussion activation function for RBF is given by:

$$\Phi(r) = \exp(-\frac{|x-c_i|^2}{2\sigma_i^2}) \tag{1}$$

Where c_i is vector value parameter centroid (first layer weight) as shown in Fig 1 and σ_i is valued shaping parameter (width). Finally, the output layer of RBF implements a weighted sum of Gaussion activation function outputs which is given as follows:

$$f(x) = \sum_{i=1}^{n} w_i \Phi(r) \tag{2}$$

Where w_i is connection weights in the second layer as illustrated in Fig. 1.

Fig. 1 shows the RBF structure that was designed with one hidden layer with two neurons, the activation function was the Gaussion function and the training algorithm was the PSO that is discussed in the next Section.

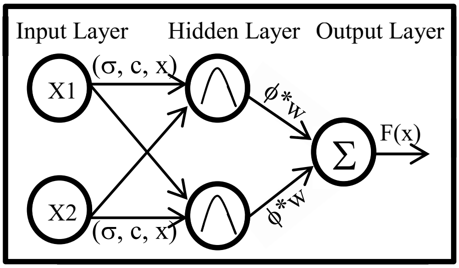


Fig. 1. RBF Structural Design

The process of the RBF consists of 6 steps. The flow chart shown in Fig 2 illustrates the steps. These steps are as follows:

i. Step 1: parameter initialization – the RBF parameters, which are c_i the vector value parameter centroid (center) and w_i the connection weight in the second layer are set to zero while σ_i the value of shaping parameter (width) is set to one.
ii. Step 2: Gaussion function – is calculated based on Eqn 1.
iii. Step 3: RBF function – is calculated based on Eqn 2.
iv. Step 4: Calculate Error – which is the difference between the actual value and the predicted value.
v. Step 5: Updating parameters – the parameters c, σ and w are updated by the following equations:

$$c = c + 2.LR.e.w.\phi.(x - c)/\sigma \tag{3}$$

Where LR is the learning rate, e is the error that was calculated is Step 4 and ϕ is the current output of Gaussion function that was calculated in Step 2.

$$\sigma = \sigma + LR.e.w.\phi.r/\sigma^2 \tag{4}$$

Where r is the distance between the input and the center of the RBF While w is updated using PSO algorithm and that is illustrated in the next Section.

vi. Step 6: Termination criterion – is represented as a number of iteration. If the termination criterion is met, the algorithm will stop otherwise Step 2 to Step 5 are repeated.

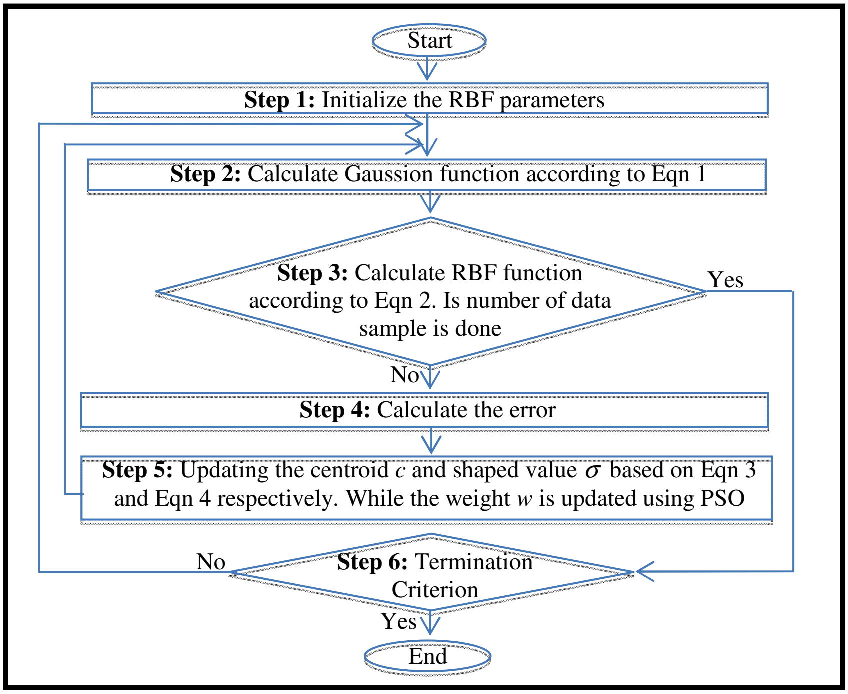

Start
Step 1: Initialize the RBF parameters
Step 2: Calculate Gaussion function according to Eqn 1
Step 3: Calculate RBF function according to Eqn 2. Is number of data sample is done
Yes
No
Step 4: Calculate the error
Step 5: Updating the centroid c and shaped value σ based on Eqn 3 and Eqn 4 respectively. While the weight w is updated using PSO
No
Step 6: Termination Criterion
Yes
End

Fig. 2. Flow Chart of RBF Process

2.2 Particle Swarm Optimization

Particle Swarm Optimization is a population based stochastic optimization technique developed by J. Kennedy and R. Eberhart in 1995. It models the cognitive as well as the social behavior of a flock of birds (solutions) which are flying over an area (solution space) in search of food (optimal solution) [23]. It is becoming very popular due to its simplicity of implementation and ability to quickly converge to a reasonably good solution [21] [6], [11]

Every single solution called a particle flies over the solution space in search for the optimal solution. Particles have a tendency to duplicate their individual past behavior that has been successful (cognition) as well as to follow the successes of the other particles (socialization). A particle status on the search space is characterized by two factors: its position and velocity, which are updated by following equations:

$$V_i\,(t+1) = w.V_i(t) + c_1.r_1(\,pbest_i - x_i(t)) + c_2.r_2(gbest_i - x_i(t)) \tag{5}$$

$$X_i(t+1) = X_i(t) + \;V_i\,(t+1) \tag{6}$$

Where w, c_1, c_2 are inertia weight and social acceleration constants respectively and r_1 and r_2 are two random numbers in the range [0, 1]. $pbest_i$ is the particle best solution and *gbest* is the global best solution, i.e. the best solution that any particle (in the whole population) has achieved. To keep the moving stability, a limited coefficient v_{max} is introduced to restrict the size of velocity which is given by the following equation:

$$| V_i (t + 1) | \leq V_{max} \tag{7}$$

Because of a large value might cause the particles to fly past good solutions, while a small number can cause the particles to get trapped in the local optima [7].

We applied an online mode of training the RBF algorithm which means the weights at each training samples are updated. A single weight of iteration and its corresponding output of the Gaussion function are produced as inputs of the PSO algorithm. In order to fit these data into PSO, we assumed that the swarm has only one particle with one individual, which is the weight. At the beginning, the velocity is initialized to current weight and the position of the particle to the current output of Gaussion function. After that, Eqn 5 and Eqn 6 are modified to fit the requirement of the proposed method. The modified equations are given as follows:

$$V_i (t + 1) = w.V_i(t) + c_1.r_1(w_i - x_i(t)) + c_2.r_2(w_i - x_i(t)) \tag{8}$$

$$W_i(t + 1) = W_i(t) + V_i (t + 1) \tag{9}$$

Where x_i is the current output of Gaussion function and *pbest* and *gbest* are replaced with the past position of the weight. Which means the particle depends on its past experience by adjusting its position according to its own experience.

3 Experiments and Discussion

We have applied our method on two well-known benchmark datasets which are Mackey-Glass Time Series (MGTS) and Competition on Artificial Time Series (CATS) and one real world dataset called Rainfall dataset. However, we first tuned the parameters of our method (RBF-PSO) using Taguchi method together with Minitab software. Based on the analysis, the prediction by Minitab and some experimental trial and error, the optimal parameters values are as follow: *LR* = 0.1, *w* = 0.3, c_1 = - 0.1 and c_2 = - 0.1. The three experiments are as follows:

3.1 Mackey-Glass Time Series Experiment

Mackey-Glass time series is generated by the nonlinear time delay differential equation called Mackey-Glass equation. The equation is given as follows:

$$\frac{dx}{dt} = \beta \frac{x_\tau}{1 + x_\tau^n} - \gamma x, \quad \gamma, \beta, n \rangle 0 \tag{10}$$

Where β, γ, τ, n are real numbers, and x_τ represents the value of the variable x at time $(t-\tau)$. Depending on the values of the parameters, this equation displays a range of periodic and chaotic dynamic.

The data size is 500 elements. The whole 500 data elements were used for training and for forecasting simultaneously. Then, the forecasted data results are compared with the actual values by calculating Normalized Rooted Mean Square Error (NRMSE) which is given by the following equation as a main standard for performance evaluation:

$$NRMSE = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left[y(i) - \hat{y}(i) \right]^2}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left[y(i) - \bar{y}(i) \right]^2}} \qquad (11)$$

The best and average results of RBF-PSO are as follows: *Best Result of NRMSE* = ***0.00157*** and *Average Result of NRMSE* = ***0.00173***. In order to have a clearer understanding, the average result was provided because of the random values that were calculated in the PSO algorithm equations as described earlier. The average result was accumulated from 30 runs.

Fig 3 shows the predicted result and actual Mackey-Glass data. When actual data and predicted data were compared, it was shown that there was not much difference and the accuracy was very good.

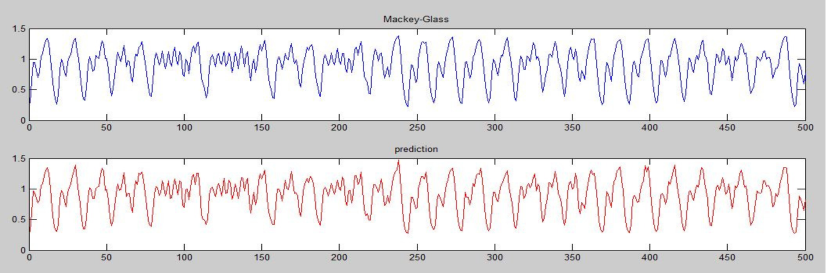


Fig. 3. MGTS data and the predicted data

Fig 4 shows the plot of the error convergence rate in terms of NRMSE. The result shows that the RBF-PSO has a very good convergence between exploration and exploitation. However, the sharp curve at the beginning in Fig 4 does not mean that the RBF-PSO has a premature convergence; this is because in the beginning the parameters that were supposed to be trained were set (initialized) to zero. After a few iterations of training, it can be clearly seen that the curve goes down gradually perfectly and has a balance between exploration and exploitation. Moreover, it can be noticed that the RBF-PSO is able to escape from falling into local minima which is one of the good characteristics of the RBF-PSO.

In order to validate the superiority of the RBF-PSO model, it was compared with several other models in the literature. Table 1 shows the Normalize Root Mean Square Error comparison method of various approaches for MGTS. The experimental results for the MGTS listed in Table 1 clearly indicate that the proposed model BFR-PSO outperformed other techniques in terms of prediction accuracy.

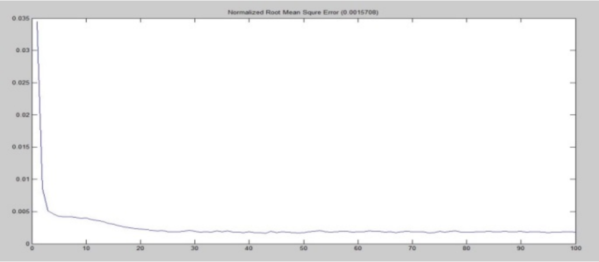


Fig. 4. The error convergence rate graph of MGTS

Table 1. The List of results of several methods on MGTS

Methods	References	Results in NRMSE
RBF-PSO	**Proposed method**	**0.00157**
Hidden Markov Model + Neural Nets	[2]	0.0017
HMMSBB + FIS	[22]	0.0018
GEFREX	[15]	0.0061
ANFIS	[8]	0.0074
EPNet	[25]	0.02
TDDFN	[13]	0.025
GFPE	[9]	0.026

3.2 Competition on Artificial Time Series Experiment

The CATS benchmark originates from the Competition on Artificial Time Series [10] organized on the IJCNN'04 conference in Budapest. Task of the predictor is to forecast five gaps in the artificial time series. The whole time series has 5000 values with the 100 missing data. The missing data are divided into five blocks as follows: 981-1000, 1981-2000, 2981-3000, 3981-4000, and 4981-5000.

The predictive error is described by two criterions: E1 and E2:

$$E1=\frac{\sum_{t=981}^{1000}(e_t-\hat{e}_t)^2}{100}+\frac{\sum_{t=1981}^{2000}(e_t-\hat{e}_t)^2}{100}+\frac{\sum_{t=2981}^{3000}(e_t-\hat{e}_t)^2}{100}+\frac{\sum_{t=3981}^{4000}(e_t-\hat{e}_t)^2}{100}+\frac{\sum_{t=4981}^{5000}(e_t-\hat{e}_t)^2}{100} \tag{12}$$

$$E\,2=\frac{\sum_{t=981}^{1000}(e_t-\hat{e}_t)^2}{80}+\frac{\sum_{t=1981}^{2000}(e_t-\hat{e}_t)^2}{80}+\frac{\sum_{t=2981}^{3000}(e_t-\hat{e}_t)^2}{80}+\frac{\sum_{t=3981}^{4000}(e_t-\hat{e}_t)^2}{80} \tag{13}$$

Where *e* is the real value of the signal $\hat{e}$ is the predicted value and *t* is the time step. The first criterion E1 describes the prediction error for all 100 missing values, while the second criterion E2 expresses the prediction error in the first four missing blocks of data (80 values). It is very important to distinguish these two criterions because some prediction methods could have problems to predict the last 20 values of the signal [10].

Similarly with MGTS, only the 100 missing data elements were used for training and 100 data elements were forecasted simultaneously. The reason why the whole 5000 data elements were not used was because it was noticed that the result did get any better by increasing the number of data to be trained. Furthermore, only the 100 missing data elements that were needed to be predicted were focused on and what was more important was to avoid computational cost.

The best and average results of RBF-PSO are as follows: *Best Result of E_1 =* ***970****,* *E_2 =* ***220*** and *Average Result of E_1 =* ***993****,* *E_2 =* ***250****.* The average result was accumulated from 30 runs.

Fig 5 shows the predicted results and the actual CATS missing data. It can be seen that the predicted results differ from the actual missing data, which means that the predictions are not accurate enough due to the natural of the data itself, which is artificial and hard to predict [10]. Another thing that can be noticed is that the difference between E1 and E2 is big; this is because of the prediction result of the fifth block was not as accurate as the rest, and most models have difficulty in predicting it [10].

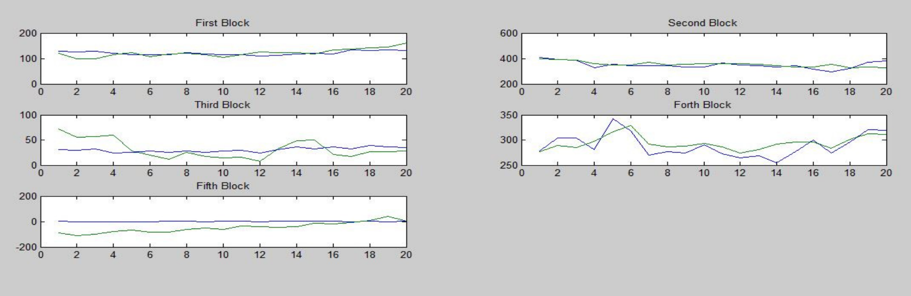


Fig. 5. The predicted data and the actual missing data of the 5 blocks. The actual data in green color while the predicted data in blue color.

In order to validate the superiority of the RBF-PSO model, it was compared with the best method in the competition and its enhanced model [28]. Table 2 shows the comparison in term of E1 and E2. It can be noticed that based on E2, we have got the best result. On the other hand, we have got competitive results in terms of E1.

Table 2. List of the experimental result for CATS

Methods	Reference	E_1	E_2
Kalman smoother with cross-validated noise density	[16]	381	312
Kalman smoother	[20]	408	346
RBF-PSO	Our method	970	220

3.3 Real World Rainfall Dataset Experiment

The real-world dataset is called rainfall dataset which is obtained from Climate Change Institute at UKM. It daily calculates the amount of rain drops in the UKM Bangi station, Selangor, Malaysia from 1979 to 2004. The rainfall dataset is an imbalanced dataset that is inherently difficult due to lack of information and it has missing values. In this section, a study is carried out on the performance of the RBF-PSO, which has gained great success on the benchmark datasets.

Two types of experiment were performed on the real-world rainfall datasets. In the first experiment, the daily data elements of the year of 2003 were used because they had less missing data elements. The missing elements were filled with the average value. In the second experiment, all the data since 1979 to 2004 were used. The total amount of the rain drops were calculated monthly to give the monthly data elements from 1979 to 2004. In the second experiment, there were no missing data elements but the data were unbalanced because they were calculated monthly and there were some missing data in each month that could not be filled with the average value. Calculations were just carried out on what was there.

In both of the experiments, the data were normalized (transformed) by making them in between the range [-1 1]. Then similarly with benchmarks datasets for both experiments, we used the whole data elements for training and we forecasted the whole data elements simultaneously. After that, the forecasted data results were compared with the actual values to form the error criterion that was used as a main standard for performance evaluation, which is Mean Square Error (MSE). The number of elements of the first experiment was 365 data elements, while the number of elements of the second experiment was 304 data elements. *First experiment result = **0.0316** and second experiment result = **0.0406**.*

Fig 6 shows the result of the first experiment (predicted results and actual rainfall data of 2003). It can be seen the predicted result was not as good as the MGTS. This is because the MGTS was determined by the delay parameter and had a limit cycle period [29], while real-world rainfall had a lot of factors that affected the prediction accuracy such as missing values and the most important unexpected natural behavior. Despite that, the RBF-PSO gave a pretty good prediction considering these factors that affected the prediction accuracy.

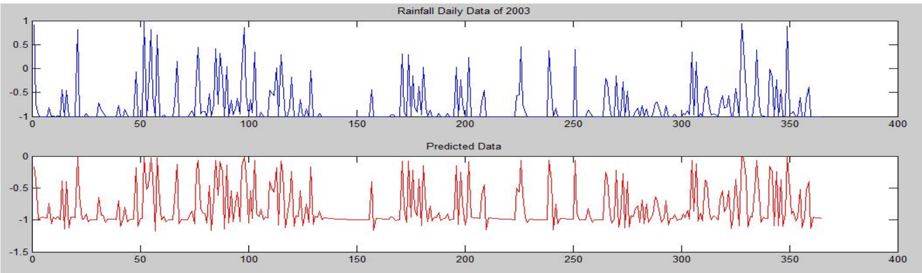


Fig. 6. The first experiment of real-world rainfall data and its prediction

Similarly, Fig 7 shows the result of the second experiment (predicted results and actual rainfall data from 1979 to 2004). It can be seen the predicted result was not as good as the first experiment. This is because the second experiment had more missing values which led to unbalanced data. Despite that, the RBF-PSO gave a pretty good prediction considering these factors that affected the prediction accuracy.

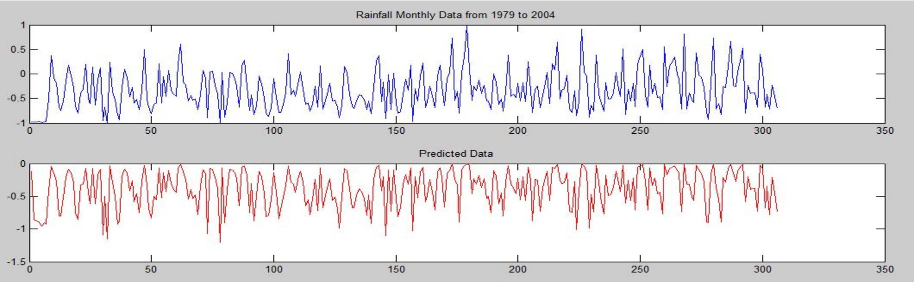


Fig. 7. The Second experiment of real-world rainfall data and its prediction

4 Conclusions

In this study, we have proposed a method (RBF-PSO) for time series prediction problems. The proposed method was able to overcome the challenges of time series prediction which are premature convergence, computational complexity and falling into local optima. The performance of the proposed method was verified by using two well-known benchmark datasets (MGTS and CATS) and a real-world dataset called Rainfall dataset. The obtained results of our proposed method have demonstrated the effectiveness and the consistency and have managed to produce a best result and competitive result when being experimented upon different kind of datasets. To sum up, we can say that the experiments that was carried out presented strong evidence and a significant contribution that has been made and become a highly appropriate methodology to be employed for time series prediction.

Acknowledgement. We would like to thank the Climate Change Institute at UKM for providing us the real world dataset (Rainfall Dataset) and facilitate our work.

References

1. Geva, A.B.: Non-stationary Time Series Prediction Using Fuzzy Clustering. IEEE (1999)
2. Saurabh, B.: Chaotic time series prediction using combination of Hidden Markov Model and Neural Nets. In: International Conference on Computer Information Systems and Industrial Management Applications (CISIM), New Delhi, India (2010)
3. Bros, A.G., Pitas, I.: Median radial basis functions neural network. IEEE Trans. on Neural Networks 7(6), 1351–1364 (1996)

4. Chen, S., Cowan, C.N., Grant, P.M.: Orthogonal least squares learning algorithm for radial function networks. IEEE Trans. on Neural Networks 2(2), 302–309 (1991)
5. Yang, C.-X., Zhu, Y.-F.: Using Genetic Algorithms for Time Series Prediction. In: Sixth International Conference on Natural Computation, IEEE Circuits and Systems Society, China (2010)
6. Eberhart, R.C., Shi, Y.: Extracting rules from fuzzy neural network by particle swarm optimization. In: Proceedings of IEEE International Conference on Evolutionary Computation, Anchorage, Alaska, USA (1998)
7. Eberhart, R.C., Shi, Y.: Particle Swarm Optimization: Developments, Applications and Resources. In: Proceedings of the 2001 Congress on Evolutionary Computation, May 27-30, vol. 1, pp. 81–86 (2001)
8. Jang, J.S.R.: ANFIS: Adaptive-Network-Based Fuzzy Inference System. IEEE Trans. Syst., Man, Cybern. 23, 51–63 (1993)
9. Kim, D., Kim, C.: Forecasting time Series with Genetic Fuzzy Predictor Ensemble. IEEE Transactions on Fuzzy Systems 5, 523–535 (1997)
10. Lendasse, A., Oja, E., Simula, O.: Time series prediction competition: The CATS benchmark. In: Proc. of IEEE Int. Joint Conf. on Neural Networks, pp. 1615–1620 (2004)
11. Li, P., Tan, Z., Yan, L., Deng, K.: Time Series Prediction of Mining Subsidence Based on Genetic Algorithm Neural Network. In: International Symposium on Computer Science and Society. IEEE Computer Society, China (2011)
12. Moody, J.: Fast learning in networks of locally-tuned processing units. Neural Computation, Bol 1, 281–294 (1989)
13. Oysal, Y.: Time Delay Dynamic Fuzzy Networks for Time Series Prediction. In: Sunderam, V.S., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2005. LNCS, vol. 3514, pp. 775–782. Springer, Heidelberg (2005)
14. Poggio, T., Girosi, F.: Networks for approximation and learning. Proc. IEEE 78(9), 1481–1497 (1990)
15. Russo, M.: Genetic Fuzzy Learning. IEEE Transactions on Evolutionary Computation 4, 259–273 (2003)
16. Särkkä, A., Vehtari, J.: CATS benchmark time series prediction by Kalman smoother with cross-validated noise density. Neurocomputing 70, 2331–2341 (2007)
17. Samek, D., Manas, D.: Artificial neural networks in artificial time series prediction benchmark. International Journal of Mathematical Models and Methods in Applied Sciences 5, 1085–1093 (2011)
18. Sanner, R.M., Slotine, J.J.E.: Gaussion networks for direct adaptive control. IEEE Trans. on Neural Networks 3(6), 837–863 (1994)
19. Sapankevych, N.I., Sankar, R.: Time Series Prediction Using Support Vector Machines: A Survey. IEEE Computational Intelligence Magazine 4, 24–38 (2009)
20. Sarkka, S., Vehtari, A., Lampinen, J.: Time Series Prediction by Kalman Smoother with Cross Validat ed Noise Density. In: IJCNN 2004, Budapest (2004)
21. Shen, H.-y., Peng, X.-q., Wang, J.-n., Hu, Z.-k.: A mountain clustering based on improved PSO algorithm. In: Wang, L., Chen, K., S. Ong, Y. (eds.) ICNC 2005. LNCS, vol. 3612, pp. 477–481. Springer, Heidelberg (2005)
22. Srivastava, S., Bhardwaj, S., Madhvan, A., Gupta, J.R.P.: A Novel Shape Based Batching and prediction approach for time series using HMM and FIS. In: Communicated in 10th International Conf. on Intelligent Systems Design and Applications, Cairo, Egypt (December 2010)

23. Hu, X., Shi, Y., Eberhart, R.: Recent Advances in Particle Swarm. In: Proceedings of the Congress on Evolutionary Computation, Portland, OR, USA, June 19-23, vol. 1, pp. 90–97 (2004)
24. Zhang, Y.-Q., Wan, X.: Statistical fuzzy interval neural networks for currency exchange rate time series prediction. Applied Soft Computing Journal 7, 1149–1156 (2007)
25. Yao, X., Lin, Y.: A new evolutionary system for evolving artificial neural networks. IEEE Transactions on Neural Networks 8, 694–713 (1997)
26. Xin, Y., Ye, Z., He, Y.-G., Hai-xia, Z.: Prediction of chaotic time series of neural network and an improved algorithm. In: 2010 IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA), Changsha, China, pp. 1282–1286 (2010)
27. Zhang, G., Patuwo, B.E., HuX, M.Y.: Forecasting with artificial neural networks: The state of the art. International Journal of Forecasting 14(1), 35–62 (1998)
28. Nicoară, E.S.: Population-Based Metaheuristics: A Comparative Analysis. International Journal of Science and Engineering Investigations 1(8), 84–88 (2012)
29. Suttorp, T., Igel, C.: In: Jin, Y. (ed.) Multi-Objective Machine Learning (2006)

Implementation of Modified Cuckoo Search Algorithm on Functional Link Neural Network for Climate Change Prediction via Temperature and Ozone Data

Siti Zulaikha Abu Bakar[1], Rozaida Ghazali[1],
Lokman Hakim Ismail[1], Tutut Herawan[2,3], and Ayodele Lasisi[1]

[1] Universiti Tun Hussein Onn Malaysia, Batu Pahat, Johor, 86400, Malaysia
gi110020@siswa.uthm.edu.my, {rozaida,lokman}@uthm.edu.my,
lasisiayodele@yahoo.com
[2] University of Malaya
50603 Pantai Valley, Kuala Lumpur, Malaysia
[3] AMCS Research Center, Yogyakarta, Indonesia

Abstract. The effect of climate change presents a huge impact on the development of a country. Furthermore, it is one of the causes in determining planning activities for the advancement of a country. Also, this change will have an adverse effect on the environment such as flooding, drought, acid rain and extreme temperature changes. To be able to avert these dangerous and hazardous developments, early predictions regarding changes in temperature and ozone is of utmost importance. Thus, neural network algorithm namely the Multilayer Perceptron (MLP) which applies Back Propagation algorithm (BP) as their supervised learning method, was adopted for use based on its success in predicting various meteorological jobs. Nevertheless, the convergence velocity still faces problem of multi layering of the network architecture. As consequence, this paper proposed a Functional Link Neural Network (FLNN) model which only has a single layer of tunable weight trained with the Modified Cuckoo Search algorithm (MCS) and it is called FLNN-MCS. The FLNN-MCS is used to predict the daily temperatures and ozone. Comprehensive simulation results have been compared with standard MLP and FLNN trained with the BP. Based on the extensive output, FLNN-MCS was proven to be effective compared to other network models by reducing prediction error and fast convergence rate.

Keywords: MLP, BP, FLNN, FLNN-MCS, Ozone, Temperature, Climate Changes, Prediction.

1 Introduction

Climate change, which has been in existence for ages, is the most important and lifelong change towards the percentage distribution in weather patterns through duration range. The reflections of the changes bring a high impact into the environment all around the world such as glaciers shrinking, breaking up of ice on rivers and lakes, plant and animal ranges shifting to conducive areas, and trees flowering sooner. On the list of the causes of climate change around the world is

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_23, © Springer International Publishing Switzerland 2014

greenhouse effect, which has a great impact on global warming relying on the increment of temperature and ozone depletion.

Prediction of climate changes in urban or local area will allow the evaluation of certain ambient ozone concentrations in urban areas with factors such as compliance and noncompliance with Environmental Protection Agency (EPA) requirements. Though ozone and temperature prediction models exist, there is still a need for more accurate models. Development of these models is difficult because the meteorological variables and photochemical reactions involved in ozone formation are complex. Previously, conventional method has been applied for temperature and ozone forecasting, and result from this method are really classy and tremendously intricate. However the algorithms via data-driven method produces more faster computations and involves less input parameter compared to process based models [1]. Hence, data driven methods deliver better alternative compared to conventional process based modelling. One of the methods applied through data driven is Neural Network (NN).

Through the architecture of network topology and link between nodes, the behavior of NN can be defined. Multilayer Perceptron (MLP), an integral part of the NN architecture, was able to generate the highest usage in network architecture. MLP consist of two or more nodes interlinked between the layers. Moreover, when the number of inputs gets higher and larger, the training of algorithm will become slower and overly dull.

Back Propagation (BP) is a supervised learning method that involves back propagation error of network weight, but with its exceptional ability, some inherent problems exist that need to be tackled. Basically, the convergence speed of BP gets slower and becomes incompatible to solve huge problems. To overcome the unbearable time problem, this research work focuses on application of FLNN [2], which contains a single layer of tunable weight that can reduce the complexity of network, error rate and increases the convergence time rate. However the convergence performance of BP algorithm highly relies on the selection of initial value of connection weights and other learning parameters. Hence Modified Cuckoo Search (MCS) was introduced and proposed to substitute BP algorithm in train the FLNN in order to reduce the prediction error and convergence time.

2 Functional Link Neural Network (FLNN)

Higher Order Neural Networks (HONNs) are expansions of first order NN [3] and models that allow higher order interactions between neurons [4]. This powerful NN has been successfully handle many problems [5]. In HONNs, there are many types of the model such as, Sigma-Pi networks, Pi-Sigma networks [6], Product Unit, Ridge Polynomial Neural Network [7] and Functional Link Neural Network. Despite the growth of various types of ANNs, HONNs [5] has proved to conveniently handle some of the problems that arise in non-HONNs.

The architecture of HONNs is attributed to Giles [5], and in conjunction with Pao [8], they introduced the functional link models called tensor network [8]. The application of HONNs in pattern recognition and associative recall task has generated successful outcomes [5]. However, only few researches have been done for time series prediction applications. Regarding the models of HONNs, the decreasing

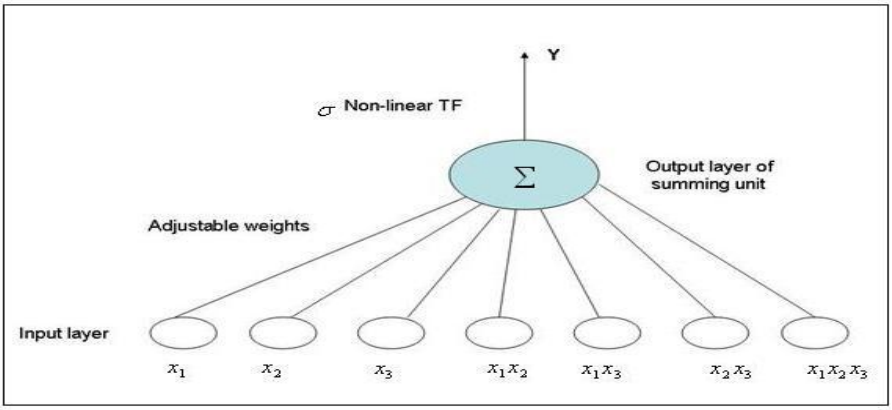


Fig. 1. Functional Link Neural Network

number of layers can solve over-fitting and local minima problems that occurred in standard ANNs [8]. Therefore, we used FLNN in order to overcome this problem [8]. Fig. 1 shows the architecture of FLNN with x_i are the input vector.

FLNN which is a type of HONN was developed by Pao [8]. It has gained wide reputation over the years. Regardless of the flexibility of training in nonlinear separable function, FLNN has successfully excelled in many application areas such as system identification [9], channel equalization [10], short-term electricload forecasting, and some of the tasks of data mining. In addition, the usability of FLNN is more powerful in handling a non-linear task compared to MLP, even though it is not carrying any hidden layers and working as a flat network compared to MLP models. The architecture of FLNN is based on a flat network which does not consists hidden layers therefore reducing complications in training the learning network.

3 Cuckoo Search (CS)

CS is one of the metaheuristic families in searching algorithm based on the cuckoo bird reproduction behavior. The aim of cuckoo algorithm is to train and reduces the error and optimise the best weight during training session .It was introduced by Yang and Deb [11] via laying their eggs in other species host bird nest. The female cuckoo bird has the ability to mimic the patterns and color of other host bird eggs and this makes the survival of their eggs higher by not been expelled from the host nest and thus increasing productivity [11].

The function of cuckoo eggs is to swap the finest solution in the other bird’s nest. The main reason of exchanging the eggs of host nest is because the cuckoo egg carries a better solution to the existing eggs in the nest. Three rules of CS were declared based on [11]:

- Only lays one egg for each nest and the egg will carry set of solution or will be abandoned.

- The new generations will bring the finest solution that is referred to the cuckoo egg
- The probability of nest is fixed either strange egg were discover by host will caused they abandon the egg or nest and create another new nest

4 Modified Cuckoo Search (MCS)

Notably, based on adequate computation, the result of CS always gives the best in finding the optimum weight [12]. However, the problem of finding the whole region via random walks still gives slow convergence. As a result, two modifications have been made to the original CS while still keeping the various algorithmic parts for a wider region of application [12]. In the first modification, changes had been made to the size of the Levy Flight step α . In CS, the value of α is constant and have been assigned a value of 1 [13]. There is an increase in the number of generations when α decreases for MCS. This same technique is also implemented in Particle Swarm Optimization (PSO) where the inertia constant is reduced with the aim of enhancing the localised search of individuals, or eggs to the nearest solution.

The second modification, on the other hand rests solely with the addition of information exchanged between the eggs to adequately minimize the convergence to its lowest. This is credited to MCS while in CS, there exist no such exchange of information and the search is performed by individual self. For details of this swarm intelligent algorithm, refer to Modified Cuckoo Search: A new gradient free optimization algorithm [12].

5 Data Collection

One of the important parts in determining the success of NN problem is data collection. The data must be validated in terms of reliability, quality, and relevance to make sure the provided data will be able to perform input output mapping. For the execution of temperature forecasting, five years historical data of daily temperature located in Batu Pahat and ranging from 1/1/2005 to 31/12/2009 were collected from Central Forecast Office, Malaysia Meteorological Department (MMD) [14] and while ozone datasets were from benchmark data from online repository.

Based on the collected data, the statistical properties of ozone and temperature data measurement is presented in Table 1. The main purpose of this task is the appraisal capability of FLNN-MCS in term of prediction of ozone and temperature measurement.

Table 1. The statistical properties of Ozone and Temperature Measurement

Datasets	Size	Average	Max	Min
Temperature	1826	26.75	29.5	23.7
Ozone	480	337.78	430	266

As a reference to Table 1, the datasets were splitted into three parts which are; 60% for training, 20% for testing and remaining 20% for the validation part.

6 Experimental Design

We have developed the algorithms using MATLAB 8.10.0 (R2010a) on Pentium® Core™^{2} Quad CPU. Result of FLNN-MCS were compared with MLP-BP and FLNN-BP where the same training, testing and validation sets are being used as shown in Tables 3 and 4.

The assessment of the prediction of ozone and temperature has been executed using the standard basis evaluation criteria such as Mean Squared Error (MSE), Normalised Mean Squared Error (NMSE), Signal to Noise Ratio (SNR) and CPU time. According to Table 2, the MLP-BP and FLNN-BP uses small value between $(0,1)$ to evade the saturation problem for all designs and insensitivity during the training process. For FLNN-MCS, the initialised values were customised within $(0.25,0.75)$ since better performance in minimising the error was found in the training process. Meanwhile for learning rate for FLNN-MCS were set of into 0.7 and number of input nodes at 5 since it give better performance result during trial and error session.

At the same time, early stopping is implemented as the stopping criteria. If the validation error increased continuously, the training of the network will be terminated. In the testing phase, the set of weight from the lowest validation error that was observed during training is employed. The minimal error was set to 0.0001 and the maximum epoch is 3000. This method is then applied for prediction of the next day value of the time series data.

Table 2. Parameters Setting for All Network Models

Network Models	Initial Setting /Weights Values	Learning Rate / Probability	Epoch/Cycle	Number of input nodes
MLP-BP	(0,1)	0.05	3000	5
FLNN-BP	(0,1)	0.05	3000	5
FLNN-MCS	(0.25,0.75)	0.7	3000	5

7 Results

Table 3 shows the MSE results for the MLP-BP, FLNN-BP and FLNN- MCS respectively. While the NMSE for the aforementioned algorithms are tabulated in Table 4.

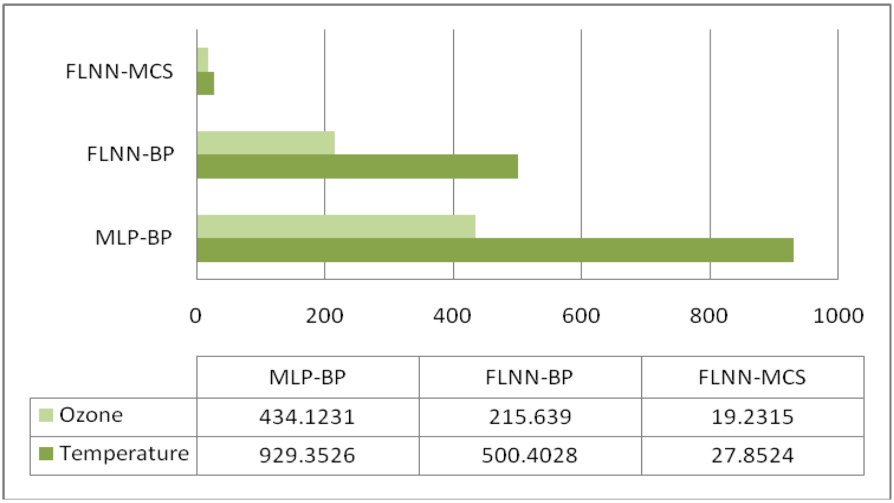
Based on the results given in Tables 3 and 4, the performance of FLNN-MCS is more outstanding compared to other models. It clearly can be seen from the MSE results of FLNN-MCS network for temperature datasets that **0.001905** is produced for higher order 3 while ozone generates **0.000101** in higher order 5 by resulting for the

Table 3. MSE for Temperature and Ozone

Datasets	TEMPERATURE			OZONE		
Hidden Layers/ Network Order	3	4	5	3	4	5
MLP-BP	0.004788	0.004776	0.004771	0.000273	0.000269	0.000277
FLNN-BP	0.004794	0.004701	0.004811	0.000312	0.000472	0.000423
FLNN-MCS	0.001905	0.001929	0.005109	0.000132	0.000106	0.000101

Table 4. NMSE for Temperature and Ozone

Datasets	TEMPERATURE			OZONE		
Hidden Layers/ Network Order	3	4	5	3	4	5
MLP-BP	0.63258	0.630973	0.630414	0.69334	0.701276	0690011
FLNN-BP	0.630791	0.630712	0.630398	0.68012	0.680015	0.680013
FLNN-MCS	0.000585	0.000948	0.000085	0.007436	0.007290	0.000683

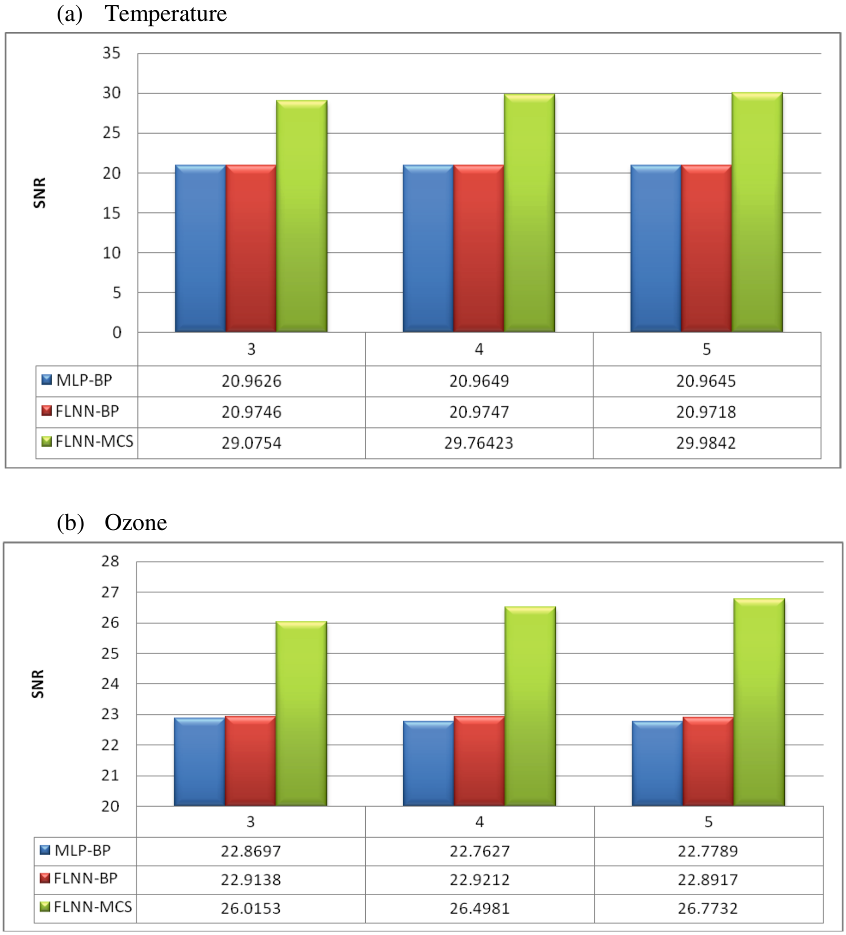


	MLP-BP	FLNN-BP	FLNN-MCS
■ Ozone	434.1231	215.639	19.2315
■ Temperature	929.3526	500.4028	27.8524

Fig. 2. Results on CPU Times

smallest error in comparison to other algorithms. For NMSE results, FLNN-MCS also proved to be slightly better than other network through results given from temperature dataset; **0.000585** and for ozone datasets; **0.000683** in the networks order. It can be concluded that the network gives the best results in the evaluation of MSE Testing and NMSE an out of sample data.

In Fig. 2, FLNN-MCS produces less convergence rate for the training phase as against the other algorithms of MLP-BP and FLNN-BP. The final result shows **27.8524s** for temperature while, **19.2315s** is recorded for ozone datasets while the values produced by MLP-BP and FLNN-BP are extremely higher than FLNN-MCS. As such, we can conclude that the algorithm of FLNN-MCS can offer the best convergence than other algorithms.


(a) Temperature

	3	4	5
MLP-BP	20.9626	20.9649	20.9645
FLNN-BP	20.9746	20.9747	20.9718
FLNN-MCS	29.0754	29.76423	29.9842

(b) Ozone

	3	4	5
MLP-BP	22.8697	22.7627	22.7789
FLNN-BP	22.9138	22.9212	22.8917
FLNN-MCS	26.0153	26.4981	26.7732

Fig. 3. SNR for Temperature and Ozone Data Time Series

Fig. 3 shows the SNR results for both data. As a conclusion, for Fig. 3 (a), FLNN-MCS results produces the higher values compared to other models, which is **29.9842** for temperature and for Fig. 3 (b) is **26.7732** for ozone data.

8 Conclusion and Future Works

Three NN models, MLP-BP, FLNN-BP and FLNN-MCS were employed and simulated for temperature and ozone time series prediction. The performance of the models is verified through MSE, NMSE, SNR and CPU time performances. From the results established in this work, it is affirmative that FLNN-MCS provides better predictions and convergence time compared to other algorithms. Therefore, it can be summarized that FLNN-MCS is competent in modeling type for temperature and ozone prediction. As for future works, we are considering test the algorithm with different datasets, in order to expand the robustness of the network model.

Acknowledgments. The author would like to thank Ministry of Higher Education for the economical support under Exploratory Research Grant Scheme (ERGS) Vote No.0882.

References

1. Paras, M.S., Kumar, A., Chandra, M.: A Feature Based Neural Network Model for Weather Forecasting in World Academy of Science Engineering and Technology (1987, 2007)
2. Giles, C.L., Maxwell, T.: Learning, invariance and generalization in high-order neural networks. In: Applied Optics, vol. 26(23), pp. 4972–4978. Optical Society of America, Washington, D.C. (1987)
3. Hopfield, J.J.: Neurons with Graded Response have Collective Computational Properties like Those of two–state Neurons. Proc. Nat. Acs. Sci. 81, 3088–3092 (1884)
4. Cohen, M.A., Grossberg, A.: Absolute stability of global pattern formation and parallel memory storage by competitive neural networks. IEEE Trans. Syst. Man. Cybern. 54, 53–63 (1986)
5. Schmidt, W.A.C., Davis, J.P.: Pattern Recognition Properties of Various Feature Spaces for Higher Order Neural Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence 15(8) (August 1993)
6. Husaini, N.A., Ghazali, R., Mohd Nawi, N., Ismail, L.H.: Jordan Pi-Sigma Neural Network for Temperature Prediction. Communications in Computer and Information Science 151(2), 547–558 (2011)
7. Ghazali, R., Hussain, A.J., Liatsis, P.: The Application of Ridge Polynomial Neural Network To Multi-Step Ahead Financial Time Series. Prediction Neural Computing and Applications 17(3), 311–323 (2008)
8. Pao, Y.: Adaptive Patten Recognition and Neural Networks. Addison-Wesley, USA (1989) ISBN: 0 2010125846
9. Patra, J.C., Bornand, C.: Nonlinear dynamic system identification using Legendre neural network. In: The 2010 International Joint Conference on Neural Networks (IJCNN), pp. 1–7 (2010)
10. Xiang, Z., Bi, G., Le-Ngoc, T.: Polynomial perceptrons and their applications to fading channel equalization and cochannel interference suppression. IEEE Transactions on Signal Processing 42(9), 2470–2479 (1994)
11. Yang, X.-S., Deb, S.: Engineering optimisation by cuckoo search. International Journal of Mathematical Modelling and Numerical Optimisation 1, 330–343 (2010)

12. Walton, S., Hassan, O., Morgan, K., Brown, M.R.: Modified cuckoo search: A new gradient free optimisation algorithm. Chaos, Solitons & Fractals 44, 710–771 (2011)
13. Yang, X.S., Deb, S.: Cuckoo search via Lévy flights, in World Congress on Nature &Biologically Inspired Computing, Coimbatore, India, pp. 210–214 (2009)
14. Climate change scenarios for Malaysia: 2001-2099. Malaysian Meteorological department scientific report (January 2009)

Improving Weighted Fuzzy Decision Tree for Uncertain Data Classification

Mohd Najib Mohd Salleh

Department of Software Engineering
Faculty of Computer Science and Information Technology
Universiti Tun Hussein Onn Malaysia
najib@uthm.edu.my

Abstract. The analytical data about the rainfall pattern, soil structure of the planting crop will partition data by taking full advantage of the incomplete information to achieve better performance. Ignoring uncertain and vague nature of real world will undoubtedly eliminate substantial information. This paper reports the empirical results that provide high return in planting material breeders in agriculture industry through effective policies of decision making. In order to handle the attribute of incomplete information, several fuzzy modeling approach has been proposed, which support the fuzziness at the attribute level. We describe a novel algorithmic framework for this challenge. We first transform the small throughput data into similarity values. Then, we propagate alternate good data and allow decision tree induction to select the best weight for our entropy-based decision tree induction. As a result, we generalize decision algorithms that provide simpler and more understandable classifier to optimally retrieve the information based on user interaction. The proposed method leads to smaller decision tree and as a consequence better test performance in planting material classification.

Keywords: Decision rules, uncertainty, fuzzy modeling.

1 Problem Definition

In supervised learning, decision modeling can be assessed by comparing model predictions to targets. Decision Support System is a computer based system utilizes model and data to support decision maker for solving unstructured problems. However, the imprecise behavior and its uncertain control required expert knowledge and effective decision making to provide meaningful decision [1,3]. We introduce weighted entropy-based decision tree induction, a novel approach to the problem of classification. Rather than finding optimal entropy of crisp data, fuzzy c-means clustering propagate alternate good data and allow decision tree induction to select the best weight to the entropy value. The imprecise information may be present in terms of fuzzy attributes to obtain exact decisions. Therefore, one of the challenges in decision tree induction is to develop algorithms which produce simple and understandable decisions [3]. As a result, the proper decision may need to adapt

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_24, © Springer International Publishing Switzerland 2014

changes in their environment by adjusting its own behavior. This research work proposes incomplete information utilizes fuzzy representation in objective function for decision tree modeling.

2 Research Background

In our study, we provide a framework to understand how the system and its components function. In our case, crop patterns and land evaluation based on physical and socioeconomic data is therefore a vital tool for decision making, since it assesses the suitability of different agro-ecological systems in an area by analyzing the relationships between the variables affecting these systems.

2.1 Uncertainty and Fuzzy Representation

In the knowledge discovery process, clustering is an established technique for grouping similar objects while separating dissimilar corresponds to the similarity of attribute values. For the normal crisp region, a point in space belongs to a certain region or otherwise (0 or 1). Since a point can now belong to boundaries instead of crisp boundaries, fuzzy set theory can be used to describe imprecise information as membership degree in fuzzy clustering, therefore a given item sets are divided into a set of clusters based on similarity [12,14]. We present fuzzy C-means based on objective function to quantify the goodness of cluster models that comprise prototypes and data partition. Fuzzy cluster analysis assigned membership degrees to deal with data that belong to more than one cluster at the same time; they determine an optimal classification by minimizing an objective function. A standard minimized objective function simply expressed as (1)

$$l = \sum_{k=1}^{p} \sum_{i=1}^{n} (\mu_k(x_i))^m \left\| x_i - c_k \right\|^2 \tag{1}$$

An iterative algorithm is used to solve the classification problem in objective function based on clustering: since the objective function cannot be minimized directly, the nearest cluster and the membership degrees are alternately optimized.

2.2 Weighted Entropy-Based Estimation

We develop decision tree using expert knowledge and additional information to generate nodes and form dynamic structure that changes when all elements remain possible to be tested [6,7,8]. All available information is applied to derive the fuzzy maximum estimation which describes the imputation of estimates for missing values in cluster centres.

Fuzzy representation in decision tree induction have been widely studied [8,9,10,11], with efforts focusing on imperfect human knowledge. In decision modeling analysis, decision tree structure refers to all the unique values which some records may contain incomplete information with discrete set of values or continuous attributes. By integration of expert knowledge and planting material, it seem reasonable to calculate the number of samples in the set Tj which belong to the same predicted class Cj assigned as (2), then the probability of a random sample of the set T belong to the class Cj .

$$freq(C_i, T) = \sum_{j=1}^{n} \mu(C_j). \tag{2}$$

With combination expert knowledge and most relevant ecological information, the membership degree can be calculated to determine some plausible data point lies to centre of the nearest cluster. We have analyzed through examples how to aggregate individual elements by considering the overall weights µ1, µ2, ...µm contribution to the decision attributes. The estimation of given attribute xi provides data in fuzzy set and gives a membership degree as in (3), as a result we calculate the individual values of attributes and aggregate the weights to obtain the final decision class in our data sets, which obtained as follow:

$$\mu = \frac{1}{m} \sum_{i=1}^{m} \mu_i x_i \tag{3}$$

2.3 Proposed Method

We propose uncertain data with different knowledge approaches in objective function for decision modeling as Weighted Entropy-based Decision tree induction (Weighted FDT). In figure 2, we describe the process of generating rules form the clustering results. We Firstly, we integrate expert knowledge and planting material data to provide meaningful training data sets using clustering approaches. Secondly, fuzzy representation is used to partition data by taking full advantages of the observed information to achieve the better performance. Finally, we optimally generalize decision tree algorithms using decision tree technique to provide simpler and more understandable models. The output of this intelligent decision system can be highly beneficial to users in designing effective policies and decision making.

The proposed method is based on possible values in the data sets, which consist of imprecise values of different data source in agricultural sectors, (see in Figure 1). The value selection assumes that the algorithm is interested in only one attribute as class as a consequent of the entropy estimation. By reducing the data, we generate different subset from selected condition attribute. This assumption significantly reduces the search space with the problem of analyzing the ambiguity of attribute values.

In decision tree, the attribute depends on its entropy computation among the rest of the attributes [15,16,17]. The entropy can be measured as the item sets are divided into subset T_j as in (4),

$$\sum_{j=1}^{n} \frac{freq(C_i, T_j)}{T_j} \log_2 \frac{freq(C_i, T_j)}{T_j}. \tag{4}$$

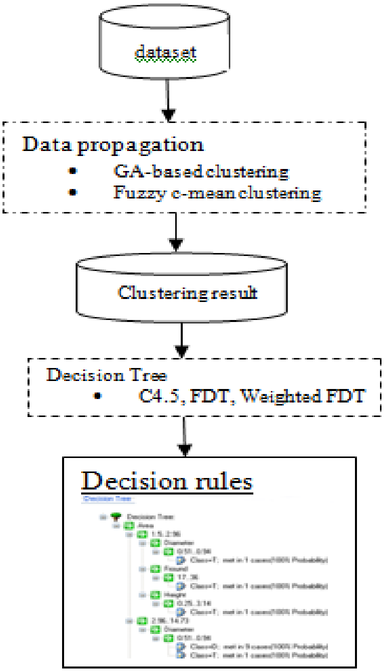


Fig. 1. The Procedure for generating decision rules from the clustering results

In the measured value x_i, the possible subset is uniquely determined by the characteristic function. The measurement of the dispersal range of value xi is relative to the true value x_o.

3 Method and Material

The effectiveness of our proposed method is demonstrated through several experiments in the environment of Matlab 7.0 on a Pentium 4 PC. Total of 1500 records of planting material had been collected during physiological analysis. These records are represented as a table of examples which described by a fixed number of features along with a predicted label denoting its class.

3.1 Crop Plantation Dataset

In previous research, to predict the evolution of interesting features in the outside environment has not been fully exploited. Decision-support in agriculture affects both crops in all their phases including planting management, costs and organization of production systems. They were also motivated by a need to integrate knowledge about soil, climate, crops and management for making better decisions. In oil palm industry, the progeny test process can be presented as a decision making process. The palm tree measurement can be considered as the condition attributes, while the selected seeds can be considered as the decision attributes. The most challenging problem is to understand the structure of the classes as dynamic evolving entities which can be affected by their environment. There is one of the reasons why most decision modeling may not perform well in data mining. We describe the special characteristic of elite palms, such as high bunch oil content, slow stem growth and short leaves were identified and selected from backcross and progeny test compact seeds. Effective breeding and selection requires a large genetic variation such as current oil palm breeding populations. Apart from targeting the primary objectives in oil palm breeding, several traits of interest include improvement of physiological traits (such as bunch index and vegetation measurement). Thus, it has become apparent to develop not only high yielding planting materials but also with novel traits. In this research study, we attempt to mine the oil palm germplasm for yield and novel traits for use in breeding and ultimately commercialization. Among others, a strategy in developing planting materials for high oil yield is through the selection against shell thickness. Fruit size is indicated by the mean fruit weight (MFW) in bunch analysis. Besides that, we incorporate vegetation measurements which provide the height, trunk diameter, frond production and leaf area per every palm tree.

4 Experimental Result

The results are presented in Table 1-3. Three UCI data sets were chosen to compare and evaluate the performance of weighted fuzzy entropy-based decision tree method (Weighted FDT) against standard C4.5 and Fuzzy Decision tree (FDT) methods. We present the statistics % MSE reduction and ROC increase averaged across the learning curves. For each data set, results were generated for 50 iterations of a random subset, fivefold cross-validation experiment. Repeated cross validation experiment was chosen to eliminate any samples selection bias created by selecting single training, pruning and testing sets. Our main goal of our evaluation is to understand and demonstrate the impact of probability estimate on the best selected seed.

4.1 Prediction Accuracy Comparison

We have studied the fuzzy based and probability estimate on data sets from UCI repository and some real-life data sets. Each data set was randomly separated 0 - 100% as training and testing set, respecting the prior of the classes. To select the proper rules set, the data sets especially continuous attributes were initially discredited using equal width binning method. We use probability estimates as evaluation criterion to select proper rules set. Based on Ziarko approach by predefined precision threshold level within the range $0.5 < \beta < 1.0$ [18].

Table 1. Test Results on Selected UCI data sets

Data Sets	Data set Sizes	β	Selected Attributes	No of Rules	Average Accuracy
Iris	150	1	2.61	6.40	0.9600
		0.9	1.29	3.55	0.9510
Breast	683	1	2.8	28.7	0.9493
		0.97	1.6	7.8	0.9238
Diabetes	768	1	4.2	98.7	0.73220
		0.8	1.5	6	0.7162
Progeny	1750	1	7	120.6	0.7340
		0.92	3	15	0.7120

In the first experiment, we take different values of β and tested 20 times for each case and present the averages of the results in Table 1, we notice that when β=1, no inconsistency allowed and shows that when a certain degree of inconsistency is allowed, the acquired rule set can become simpler without significant decreases on classification accuracy.

4.2 The Performance Measurement

In the second experiment, we show the result of maximum confidence of seed propagation. The ability to recover uncertain samples for improving classification model is important, however, estimating a confidence measure associated with sample classification may be more useful. The relationship found between accuracy classification and maximum confidence of classification is encouraging for the use of fuzzy c-means clustering compare to GA iteration. We select neighbours according to whose similarity exceeds a certain threshold value. We vary the size of threshold value and compute the MAE. The formal definition of MAE is as follows.

$$MAE = \sum_{i=1}^{n} \frac{|\, p_i - q_i \,|}{N}. \tag{6}$$

N is the number of ratings in the testing set, qi is real rating and pi is predicted.

Figure 2 illustrate weighted FDT converges faster than GA method. We use scatter plot to show the required convergence iterations. The coordination of each point is the iterations of weighted Fuzzy C-means clustering (x-axis) and Genetic Algorithm (y-axis). There 30 data points in the plot. A data point lays in the lower triangle showed that weighted FDT converges faster than GA.

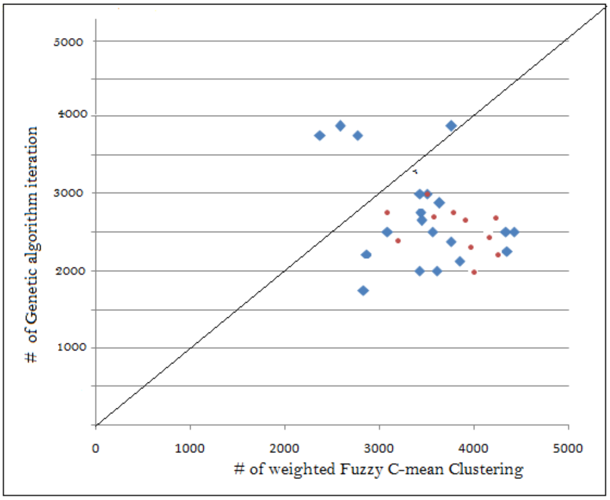


Fig. 2. Training data of physiological analysis with GA and Weighted Fuzzy C-mean clustering

Next, we performed more effective decision tree classifier by selecting suboptimal input variables in decision learning. Therefore, we select only look at feature subsets from input variables. In the process of training decision trees on bootstrap replicas of input data, we select the best split of suboptimal input variables to improve the predictive power of the ensemble and reduce correlation between trees in the ensemble.

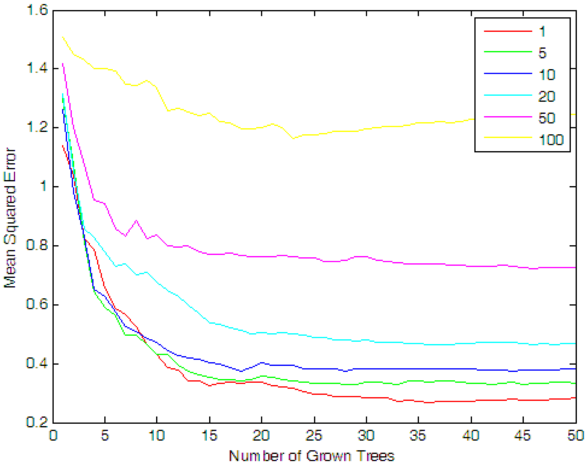
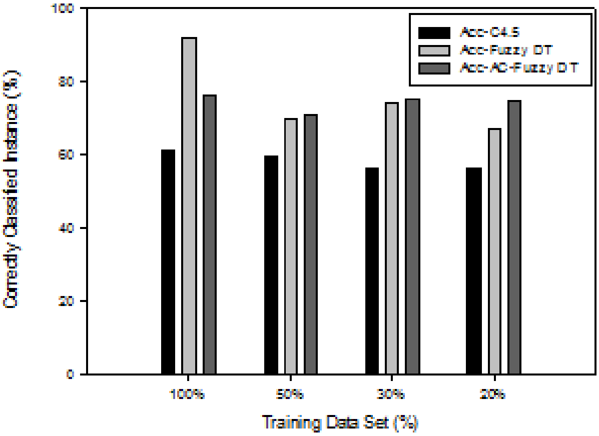


Fig. 3. Decision tree grown with mean squared error of Classification by suboptimal input variables selection

Several features of bagged decision trees make them a unique algorithm. Drawing N out of N observations with replacement omits on average 37% of observations for each decision tree. These are “out-of-bag” observations which are used to estimate the predictive power and feature importance. In figure 3, we shows the result of each observation, we estimate the out-of-bag prediction by averaging over predictions from all trees in the ensemble for which this observation is out of bag and then compare the computed prediction against the true response for this observation.

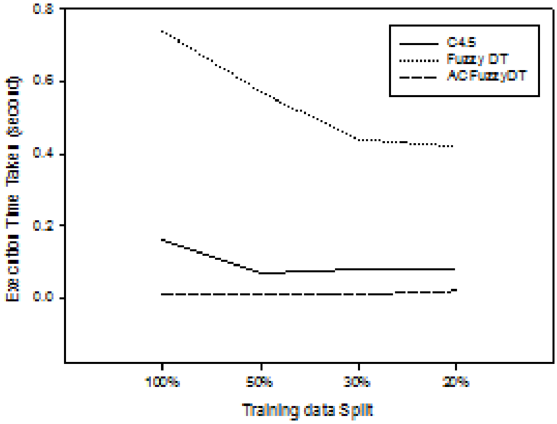
By comparing the out-of-bag predicted responses against the true responses for all observations used for training, the average out-of-bag error is estimated. We also obtain out-of-bag estimates of feature importance by randomly permuting out-of-bag data across one variable or column at a time and estimating the increase in the out-of-bag error due to this permutation. The larger they increase, the more important the feature and obtain reliable estimates of the predictive power and feature importance in the process of training, which is an attractive feature of bagging. The results of next experiment is summarized in Table 2 and Table 3, show the comparison of the accuracy (or number of correctly classified instances) and learning time (or time taken to build the model) on the dataset between C4.5, FDT and Weighted FDT.

Table 2. Correctly Classified Instances vs Training Split Data



The relative performances offer some guidance in deciding on which percentage split should be the optimal choice. From our results, weighted FDT shows the best result in term of accuracy at 50% split and gradually decreases at the stage of 70% training data sets. However, the characteristics of fast learning Weighted FDT algorithm achieves the highest accuracy when they need percentage split between 50% and 70% to. J48, on the other hand, needs all the training data to reach the highest accuracy rate.

Table 3. Execution Time Taken versus Training Data Split



4.3 Decision Rules

The result of decision tree shows some of the rule production which obtained from Estard Data Miner application (see Fig.7). From the experiment, we execute the program and obtained two decision class with each gene types, one consisting of those palm seed with ‘D’ Dura type and ‘T’ Tenara type. Using the best feature selection and attempt to create indistinguishable rules at this point, we generated all the possible rules from the decision table and then removed those that contradicted each other from the ‘T' type and 'D' type set. By this means we prevent the production of spurious rules, while still producing a reasonable number of rules from a small decision table.

The analysis uses the trait set to produce good rules and is examined to see if any, and the contradicted rules are removed from the final rule sets. This is done by removing all unknown or missing values set with the same parameters that have equal or worse values than the original rule. We extracted the relevant parameters for each of these trait under their physiological analyse. Those traits induced for reasons other than our indications were removed from the database. We then calculated the ‘actual’ rate of induction for the relevant indications. The rules obtained above are then applied to the relevant database. Experimental results have shown that our proposed method produces relatively small sized and comprehensible trees with high accuracy in generalisation and classification. It improves the performance of existing classifier in terms of both generalisation accuracy and particularly classification accuracy. Our proposed Weighted FDT also outperforms C4.5 and FDT in terms of tree size and classification accuracy. However, this method’s generalisation accuracy remains lower than that of C4.5.

5 Conclusion

In this paper, we addressed the problem of decision modelling from small sets of uncertain information and proposed a novel measure of weighted entropy-based induction to provide most simple and understandable decisions.

Our contribution is formulating an estimation of the degree of fuzzy membership which provides relatively high levels of uncertain information with small testing set. We show that the weighted entropy-based fuzzy decision tree can be used for small samples and produces better result with integration of expert knowledge. However, this method still produces fairly large error after pruning with more missing values.

In the experiments using artificially propagated data, the proposed method as more effective and showed very promising result in overall classification accuracy in these domains.

Acknowledgement. The author would like to thank Universiti Tun Hussein Onn Malaysia for funding the research grant 0824.

References

1. Chao, S., Xing, Z., Ling, X., Li, S.: Missing is Useful: Missing Values in Cost-Sensitive Decision Trees. IEEE Transactions on Knowledge and Data Engineering 17(12) (2005)
2. Schafer, J., Graham, J.: Missing data: Our view of the state of the art. Physchological Methods 7, 147–177 (2002)
3. Rouse, B.: Need to know - Information, Knowledge and Decision Maker. IEEE Transaction on Systems, Man and Cybernatics-Part C: Applications and Reviews 32(4) (2002)
4. Passam, C., Tocatlidou, A., Mahaman, B.D., Sideridis, A.B.: Methods for decision making with insufficient knowledge in agriculture. In: EFITA Conference, Debrecen, Hungary, July 5-9 (2003)
5. Rokach, L., Maimon, O.: Top-Down Induction of Decision Tree Classifiers-A Survey. IEEE Trans on System, Man and Cybernatics-Part C: Application and Reviews 35(4) (2005)
6. Quinlan, J.R.: Induction of Decision Trees. Machine Learning 1, 81–106 (1986)
7. Quinlan, J.R.: Improved use of continuous attributes in C4.5. Journal of Artificial Intelligence Research 4, 77–90 (1996)
8. Yuan, Y., Shaw, M.J.: Induction of fuzzy Decision Trees. Fuzzy Sets System 69, 125–139 (1995)
9. Janikow, C.Z.: Examplar Learning in Fuzzy Decision Tree. In: Proceedings of International FUZZ-IEEE, pp. 1500–1505 (1996)
10. Klir, G.J., Yuan, B.: Fuzzy Sets and Fuzzy Logic, Theory and Application. Prentice Hall PTR, Englewood Cliffs (1995)
11. Chiang, I.J., Hsu, Y.J.: Fuzzy classification trees for data analysis. Fuzzy Sets and Systems, International Journal in Information Science and Engineering 130(1), 87–99 (2002)
12. Heiko, T., Christian, D., Rudolf, K.: Different Approaches to Fuzzy Clustering of Incomplete Datasets. International Journal of Approximate Reasoning, 239–249 (2004)

13. Tokumaru, M., Muranaka, N.: Product-Impression Analysis Using Fuzzy C4.5 Decision Tree. Journal of Advanced Computational Intelligence and Intelligent Informatics 13(6) (2006)
14. Wang, X., Borgelt, C.: Information Measures in Fuzzy Decision Tree. In: Fuzzy Systems, Proceedings and IEEE International Conference (2004)
15. Marques, J.: Tree Classifiers Based on Minimum Error Entropy Decisions. Canadian Journal on AI, Mach Learning & Pattern Recognition 2(3) (2011)
16. Olaru, O., Wehenkel, L.: A Complete Fuzzy Decision Tree Technique. Fuzzy Sets and Systems 138, 221–254 (2003)
17. Dietterich, T., Kearns, M., Mansour, Y.: Applying the weak learning framework to understand and improve C4.5. In: 13th International Conference on Machine Learning, pp. 96–104 (1996)
18. Ziarko, W.: Set approximation quality measures in the variable precision rough set model. In: Abraham, A., Ruiz-del-Solar, J., Koppen, M. (eds.) Soft Computing Systems: Design, Management and Applications, pp. 442–452. IOS Press (2002)

Investigating Rendering Speed and Download Rate of Three-Dimension (3D) Mobile Map Intended for Navigation Aid Using Genetic Algorithm

Adamu I. Abubakar[1], Akram Zeki[1], Haruna Chiroma[2], and Tutut Herawan[3,4]

[1] Department of Information Systems
International Islamic University Malaysia
Gombak, Kuala Lumpur, Malaysia
100adamu@gmail.com, akramzeki@iium.edu.my
[2] Department of Artificial Intelligence
[3] Department of Information systems
University of Malaya
50603 Pantai Valley, Kuala Lumpur, Malaysia
[4] AMCS Research Center, Yogyakarta, Indonesia
freedonchi@yahoo.com, tutut@um.edu.my

Abstract. Prior studies have shown that rendering 3D map dataset in mobile device in a wireless network depends on the download speed. Crucial to that is the mobile device computing resource capabilities. Now it has become possible with a wireless network to render large and detailed 3D map of cities in mobile devices at interactive rates of over 30 frame rate per second (fps). The information in 3D map is generally limited and lack interaction when it's not rendered at interactive rate; on the other hand, with high download rate 3D map is able to produce a realistic scene for navigation aid. Unfortunately, in most mobile navigation aid that uses a 3D map over a wireless network could not serve the needs of interaction, because it suffers from low rendering speed. This paper investigates the trade-off between rendering speed and download rate of the 3D mobile map using genetic algorithm (GA). The reason of using GA is because it takes larger problem space than other algorithms for optimization, which is well suited for establishing fast 3D map rendering speed on-the-fly to the mobile device that requires useful solutions for optimization. Regardless of mobile device's computing resources, our finding from GA suggest that download rate and rendering speed are mutually exclusive. Thus, manipulated static aerial photo-realistic images instead of 3D map are well-suited for navigation aid.

Keywords: Rendering speed, 3D dataset, Download rate, Genetic algorithm.

1 Introduction

The emergence of 3D maps for mobile devices which can be used for navigation aid comes as the result of the drawbacks perceived with the conventional two-dimension

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_25, © Springer International Publishing Switzerland 2014

(2D) map. It's now possible to render large and detailed 3D map of cities in mobile devices at interactive rates of over 30 frame rate per second (fps) [1]. However, with increasing in mobile device's computing resources, high rate of rendering 3D maps is possible. The development of mobile 3D applications was long hindered by the lack of efficient mobile platforms [2], this affects 3D rendering interfaces. However, the growing support for Graphics processing unit's capabilities of different devices, and with the mobile devices' support for wireless network technology, paves way to running large complex 3D application in devices especially smart phones. Unfortunately, despite these opportunities and with the increase availability of wireless connectivity, yet there is still a problem with the trade-off of speed of rendering 3D dataset on-the-fly to the mobile in a real-time. Most mobile devices that use 3D map for navigation aid nowadays lack the ability to display attractive virtual world. The intended interpretation of the term 3D use in this paper does not focus on pictorially realistic satellite imagery (see Fig 1) that allows 3D view manipulation, as it's the most use for many devices which claim to be using 3D maps for navigation aid. However, this meaning is very different to alternative, and equally valid, interpretations of the term 3D, such as the 3D views provided by in-car navigation aids such as TomTom's map view (See Fig 2).

Such tools typically use a 3D projection of upcoming road layout, without substantial pictorial realism, and without freedom for the user to substantially manipulate the viewpoint. There are many other examples of systems that apply different constraints and presentation means for the purpose of map navigation. For that reason, the 3D map that this paper is referring is a 3D map model representation of a physical environment (see Fig 3), emphasizing the 3D characteristics [1-2].

Therefore, we propose to establish a trade-off for rendering speed and download rate using GA in order generalized a sequence of 3D map dataset view suited for mobile device on-the-fly without delay. The reason of using GA is because it takes larger problem space than other algorithms for optimization, which is well suited for establishing fast 3D map rendering speed on-the-fly to the mobile device that requires useful solutions for optimization. To the best of our knowledge, this approach has not been used to solve similar problems.

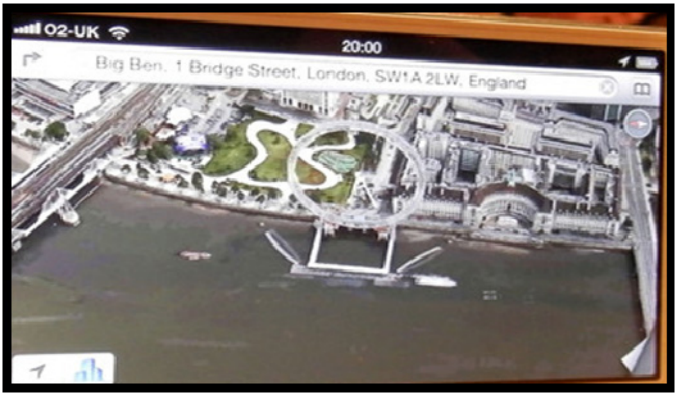


Fig. 1. Pictorially realistic satellite imagery that allows 3D view manipulation [3]

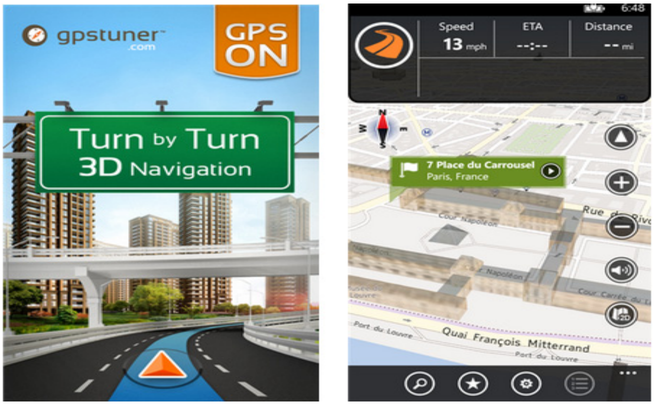


Fig. 2. 3D projection of upcoming road layout, (left) and navigation details by the right [4]

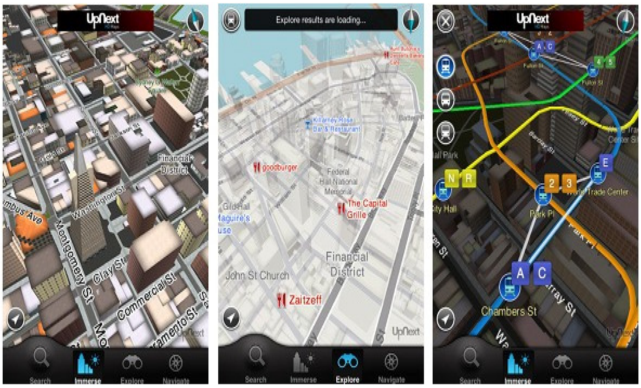


Fig. 3. 3D map model representation emphasizing the 3D characteristics [5]

Following this section, the paper is organized as follows. Section 2 presents Rendering and Download rate for 3D mobile Map, followed by section 3 which presents GA applications in the context of this work. Section 4 is the simulation result and section 5 is the conclusion.

2 Rendering and Download Rate for 3D Mobile Map

Generally, the term 3D is applied to an object that has three dimensions of measurement: Width, Height, and Depth. The whole point of 3D representation is the third dimension [6], but since the screen is only two-dimensional, the third dimension appears only indirectly in terms of perspective and occlusion. The eye is at the origin of 3D objects presentation, which can be transformed relative the distance of the objective, refers to as perspectives. In 3D representation of an object where x, y, and z represent the coordinates of the object respectively. It is proven by [6], that the homogeneous per displayed matrix when it is display in a secrete observer to perspective, observer will perceive it in the following equation:

$$[\hat{x}\ \hat{z}\ \hat{y}] = [\hat{x}\ \hat{z}\ 1]\begin{bmatrix} a & 0 & 0 \\ 0 & b & d \\ 0 & c & 0 \end{bmatrix} = [ax\ bz + c\ dz] \tag{1}$$

The 3D objects are more complex to handle than other multimedia data, such as audio signals or 2D images, since there are many different representations of such objects. For instance, a 2D image has a unique and rather simple representation: a 2D grid $(n \times n)$ composed of n^2 elements (named pixels) each containing petc.uce digital images (digital cameras, scanners, and etc.) all provide the same representation. However, for 3D models there are different kinds of representation: an object can be represented on a 3D grid like a digital image, or 3D Euclidean space. In the latter case, the object can be expressed by a single equation (like algebraic implicit surfaces), by a set of facets representing its boundary surface or by a set of mathematical surfaces. The main difficulties with a 3D object are that: The different sources of 3D data do not produce the same representations and the different applications do not consider the same representations [7-8]. Moreover, Changing from one representation to another is quite complex and often constitute open problems.

The rendering of 3D map (see Fig. 4) to a client mobile device simply start from the server side and it's described as out-of-core rendering [8]. Therefore, downloading complex 3D models with low-end devices like mobile devices over low Internet connection could be difficult.

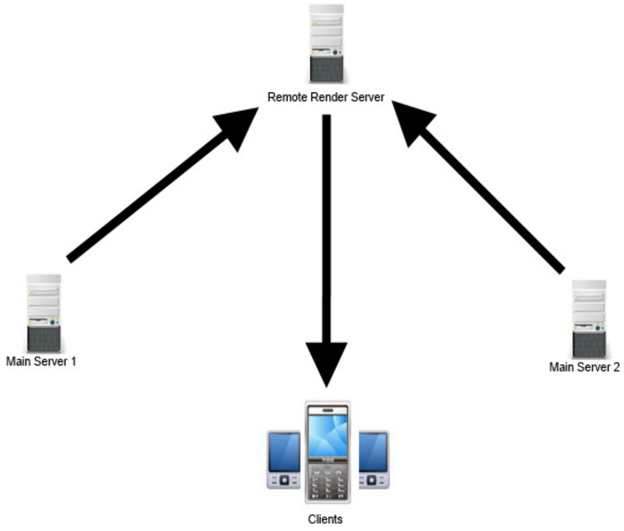


Fig. 4. The remote rendering architecture

Unfortunately, in some cases, the download/rendering rate fluctuates even with a high-end devices like mobile devices over the high Internet connection, because it is expected where there is high internet speed downloading complex 3D model will be faster, and if the GPU support in the client mobile device is huge, then rendering should also be very fast. Its observe latencies and scalability problems are sometimes the major cause of fluctuation of rendering [2]. The minimum latency is the round-trip-time of the network, and the transmission time for the payload. The result of an experiment carried out in [2] with a 20kB average frame size and 40kB/s network speed, the latency is 150ms + 500ms = 650ms, without the contribution of rendering and encoding.

In this research a simple experimental observation was carried out in order to determine the download rate and rendering speed. The server is the workstation situated in the lab with Intel Core 2 Duo 1066MHz and 5754 Gigabit Ethernet LAN with, 533MHz (4-4-4 latency), 667MHz (5-5-5 latency), and 800MHz (6-6-6 latency). Six mobile clients are used with 480 x 800 pixels, 4.3 inches (217 ppi pixel density), and the rest are below this specification. The average Download rate (kbps) and Rendering rate (fps) obtained for rendering of a 3D dataset from the server to the mobile client during 63 different periods were 42.28 kbps and 23.49 fps respectively. The gathered experimental data on the download rates and rendering speed from 63 periods (see Fig. 5) are used for optimizing the rendering and download speed through genetic algorithms.

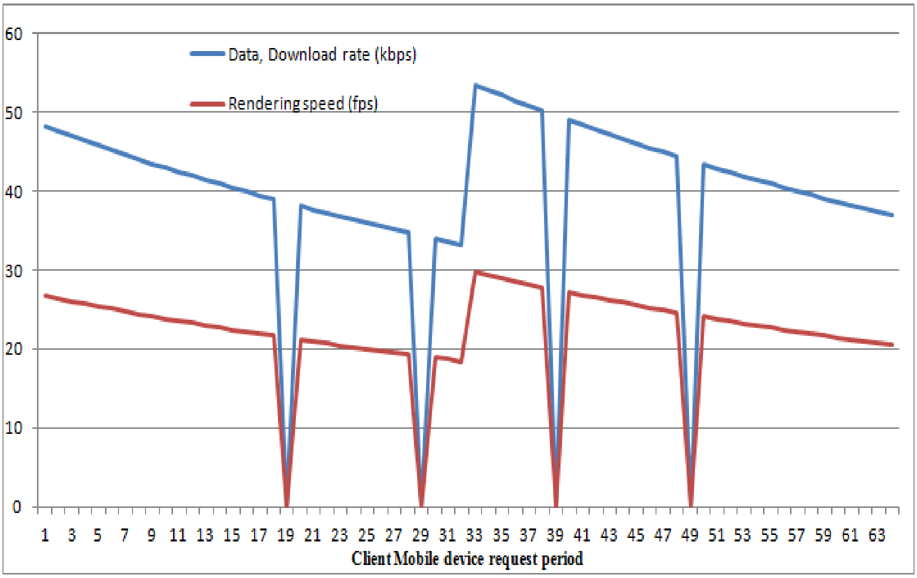


Fig. 5. The gathered experimental data on the download rates and rendering speed from 63 periods

3 Genetic Algorithm

The illustration of the Darwinian theory of evolution was represented in the form of an algorithm by [9]. The algorithm is widely accepted for solving varying real life problems in various domains. This algorithm is referred to genetic algorithm due to its biological inclination as regard to the inspirational origin. The algorithm operates by generating an initial population of individuals and encoded to compete with one another based on a fitness function as a measure of performance criterion. During the competition, some individuals are eliminated for lack of high value of the fitness function, whereas those with high values of the fitness function survive to the next generation through crossover and mutation. The crossover is operated by selecting a binary bit string for the two pair of mates and swapped. In the mutation operation, new material is introduced into the pool of gene in order to prevent the genetic algorithm from being trapped in local minima. Strings are mutated in a process of bit. The new generated populations are used for further searches by the genetic algorithm operations, where the evolution continues until a stoppage point is reached where the best solution emerged, by testing for conditions initially set as termination criterion. If the condition is satisfied, the best individual within the existing population is returned as the optimal solution [11-12]. The paper utilizes genetic algorithm to study 3D map rendering speed and data rate.

3.1 Formulating of the Objective Function and the Optimization Processes

In this section, we describe the steps required to formulate our objective function and find the relationships between download rate and rendering speed, so as to use a genetic algorithm to optimize the rendering and download speed of the mobile device [13]. We defined x_k as the rendering speed of the k-th processor in the 3D mobile device, let w_{jk} denote the amount of data type j in one unit of 3D mobile device being downloaded, defined as m_j for the 3D mobile device. Since the objective is to maximize the rendering speed, then, we formulated the problem as:

$$\text{Maximize } y = \sum c_i x_k ,$$

$$\text{Subject to } \text{Lbound} \le \sum c_i x_k \le \text{Ubound} , \ \forall x_j \ge 0 .$$

In our problem formulation, the upper (Ubound) and lower (Lbound) bound can be assigned a sufficiently large value and zero value, respectively. So that's enough space can be searched for the maximum rendering speed required to deliver a file on the mobile device. It can be noted in the objective function that all the constraints are nonnegative, that is because the rendering speed only deals with positive values as we don’t have negative rendering speed. To minimize the download rate, the expression is given as:

$$\text{Lbound} = \frac{1}{2}\left(a_{jk} - y_i\right)\left(d_{jk} \le \sum a_{jk} x_k \le \text{Ubound}\right) . \tag{2}$$

The mathematical algorithm is originally presented in [14] and we modified in the context of our studies and presented as follows: The maximization and minimization problem is encoded as

$$Lc = \log 2 \frac{\sum c_j x_k}{r} . \tag{3}$$

where c_j is chromosome, Lc = length of chromosome, and r = accuracy. Suppose n, y_j and $y_i^{'}$ are training exemplars, target and predicted values respectively, then fitness function is computed as follows:

$$f\left(s_j\right) = \frac{1}{2}\sum_{i=1}^{n}\left(y_j - y_i\right)\left(\text{Lbound} \le \sum a_{jk} x_k \le \text{Ubound}\right), \tag{4}$$

where s_j is the j-th string. Suppose G is a current group, n randomly selected and f_n independent individuals as the parent group, fitness value of B_i is $f\left(B_i\right)$, $B_i \in G$ and probability of B_i is as follows.

$$P_s\left(B_i\right) = \frac{\ell^{\int \frac{(B_i)}{T_k}}}{\sum_{B_i \in G} \ell^{\int \frac{(B_i)}{T_k}}} \sum C_j x_k , \tag{5}$$

where T_k is annealing temperature tending to 0 and

$$T_k = \iiint T_0 \frac{(\ln 2)^{\alpha-1}}{\left(\ln(1+k)\right)^{\alpha-1} k^{2-\alpha}} dT_0 ,$$

where $\alpha = 1,2$ and $k = 1,2,3,\cdots$. The value of α set to 1, $T_o = 100$ and α is selected by stem. Mutation probability $\left(P_m\right)$ is computed as follow.

$$P_m = \begin{cases} P_m - \dfrac{\left(P_{m1} - P_{m2}\right)\left(f - f_{avg}\right)}{f_{\max} - f_{avg}}, & \text{for } f \ge f_{avg} \\ P_{m1} & , \text{for } f < f_{avg} \end{cases} , \tag{6}$$

where f is the rate of fitness for individuals in possible mutation. Suppose $f_{\max}$ - largest fitness in a group, f_{avg} average fitness value of every group of generation, f larger value of fitness for two individual to be crossed and p_c is computed as follow.

$$P_c = \begin{cases} P_{c1} - \dfrac{(P_{c1} - P_{c2})(f - f_{avg})}{f_{\max} - f_{avg}}, \text{ for } f \geq f_{avg} \\ P_{c2} \qquad\qquad\qquad\qquad\quad , \text{ for } f < f_{avg} \end{cases}, \qquad (7)$$

where p_c is the crossover probability, p_{c1} less than initial rate of crossover p_{c2}.

This procedure is iteratively repeated until the optimal individual emerges as the solution to the problem. In our case the maximum and minimum values since it's an optimization problem.

4 Results and Discussion

The results indicated that the best fitness occurred at 131 generations with mean square error (MSE) of 0.001527 (see Fig. 6) and the best average fitness occurs in the 200 generation with a minimum MSE of 0.000253 (see Fig. 7). The MSE of the best average MSE indicated an improvement over the best MSE. This was expected because the ensemble of various results is typically better than the result emanated from individual algorithm. Therefore, the convergence follows the expected pattern.

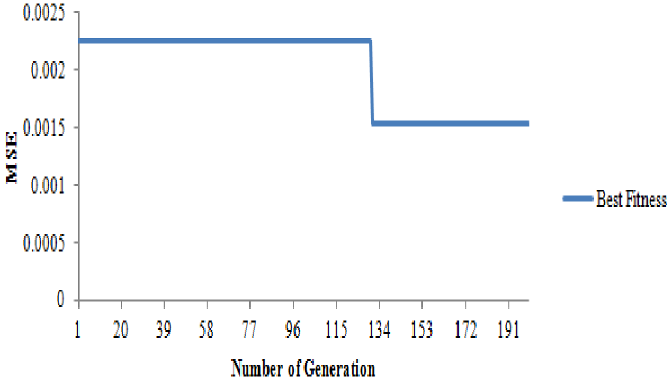


Fig. 6. Best fitness (MSE) vs. generation

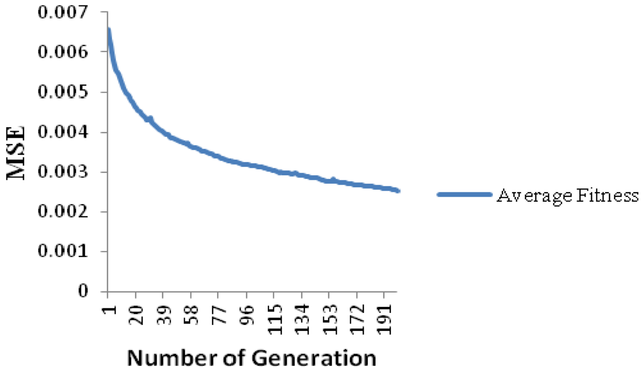


Fig. 7. Average fitness vs. generation

The genetic algorithm search was implemented using Neuro-Solution. The population size was set to 50, crossover rate was set to 0.6 and mutation rate was set to 0.001 which were the standard settings of genetic algorithm operation parameters widely used by the evolutionary community. Roulette-wheel was the selection function used in this research work. The criterion use for validating the result is the closeness of the optimization values suggested by the genetic algorithm to rendering rate of 30 fps and download rate of 100kbps. The optimized values of rendering and download speed to be expected presented in Fig. 8 with satisfactory performance as clearly indicated.

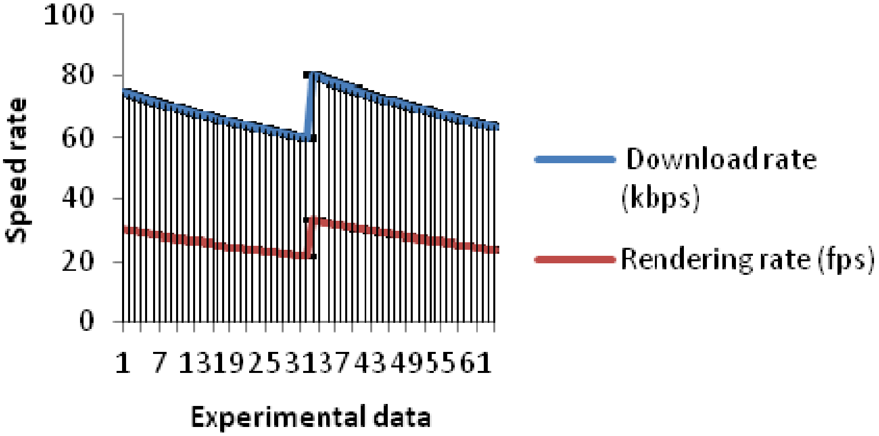


Fig. 8. Genetically optimized download and rendering rate

5 Conclusion

This paper presents an optimization of 3D mobile map rendering speed and download rate quality using genetic algorithm. A 3D model for mobile device just as any other 3D graphics rely on a graphics processing unit (GPU) which is dedicated hardware meant for reading rendering 3D objects. 3D rendering involves in-core and out-of-core rendering. Thus 3D model rendered in workstations, personal computers, or mobile devices are within in-core rendering. Whereas remote rendering is an out-of-core rendering, which involves the situation where 3D dataset viewpoint are rendered on-the-fly and in real-time. The main problem lies within the remote rendering speed and the download rate of the 3D dataset. This paper investigates the trade-off between rendering speed and download rate of the 3D mobile map using GA.

The reason of using GA is because it takes larger problem space than other algorithms for optimization. The download rate, and rendering speed were gathered for about sixty four times (period). The gathered experimental data on download rates and rendering speed from the 64 periods are used in building a GA model in order to establish a trade-off for the download rate and rendering speed. Our finding suggests that download rate and rendering speed are mutually exclusive. Thus, manipulated static aerial photo-realistic images instead of 3D map are well-suited for navigation aid.

Acknowledgement. This work is supported by University of Malaya.

References

1. Nurminen, A.: m-LOMA-a Mobile 3D City Map. In: Proceedings of the Eleventh International Conference on 3D Web Technology, pp. 7–18 (2006)
2. Nurminen, A.: Mobile 3D City Maps. Journal of IEEE Computer Graphics and Applications 28(4), 20–31 (2008)
3. Roth, A.: iPhone 5 vs. iPhone 4S EARLY VIEW Is an upgrade from the iPhone 4S to iPhone 5 worth the money?, http://www.techradar.com/news/phone-and-communications/mobile-phones/iphone-5-vs-iphone-4s-1096421 (retrieved December 20, 2012)
4. MBGW. Turn by Turn Navigation 3D App Featuring Offline 3D Maps Now Available for Windows Phone 8, http://www.wp7connect.com/2013/01/05/turn-by-turn-navigation-3d-app-featuring-offline-3d-maps-now-available-for-windows-phone-8 (retrieved December 20, 2012)
5. Seth, C.: Amazon Buys UpNext, Foreshadows Escalating 3D Map Wars, http://hothardware.com/News/Amazon-Buys-UpNext-Foreshadows-Escalating-3D-Map-Wars-/ (retrieved December 20, 2012)
6. B.J.F.: Jim Blinn's Corner: W Pleasure, W Fun. Journal of IEEE Computer Graphics and Applications Computer 18(3), 78–82 (1998)
7. Dugelay, J., Baskurt, A., Daoudi, M.: 3D Object Processing Compression, Index-ing and Watermarking, p. 210. John Wiley & Sons, West Sussex (2008)
8. Watt, A.: 3D Computer Graphics, 3rd edn., p. 624. Pearson Education, Edinburgh (2000)

9. Hesina, G., Schmalstieg, D.: A Network Architecture for Remote Rendering. Technical Report TR-186-2-98-02, Institute of Computer Graphics and Algorithms, Vienna University of Technology, Vienna, Austria (1998)
10. Teddy, M., Abubakar, A., Haruna, C.: Pedestrian position and pathway in the design of 3D mobile interactive navigation aid. In: Proceedings of the 10th International Conference on Advances in Mobile Computing & Multimedia (MoMM), pp. 189–198. ACM Press (2012)
11. Holland, J.H.: Adaptation in natural and artificial systems. MIT Press, Massachusetts (reprinted in 1998)
12. Zhang, D., Zhou, L.: Discovering Golden Nuggets: Data Mining in Financial Application. IEEE Transaction on System Man and Cybernetics—Part c: Appl Reviews 34(4), 513–522 (2004)
13. Haruna, C., Abdul-Kareem, S., Abubakar, A.: A Framework for Selecting the Optimal Technique Suitable for Application in a Data Mining Task. Future Information Technology, 163–169 (2014)
14. Ai-Ping, J., Feng-wen, H.: Methods for optimizing weights of wavelet neural network based on adaptive annealing genetic algorithm. In: 16th International Conference on Industrial Engineering and Engineering Management, pp. 1744–1748 (2009)

Kernel Functions for the Support Vector Machine: Comparing Performances on Crude Oil Price Data

Haruna Chiroma[1], Sameem Abdulkareem[1],
Adamu I. Abubakar[2], and Tutut Herawan[3,4]

[1] Department of Artificial Intelligence
[3] Department of Information systems
University of Malaya
50603 Pantai Valley, Kuala Lumpur, Malaysia
[2] Department of Information system
International Islamic University
Gombak, Kuala Lumpur, Malaysia
[4] AMCS Research Center, Yogyakarta, Indonesia
freedonchi@yahoo.com, 100adamu@gmail.com,
{sameem,tutut}@um.edu.my

Abstract. The purpose of this research is to broaden the theoretic understanding of the effects of kernel functions for the support vector machine on crude oil price data. The performances of five (5) kernel functions of the support vector machine were compared. The analysis of variance was used for validating the results and we take additional steps to study the Post Hoc. Findings emanated from the research indicated that the performance of the wave kernel function was statistically significantly better than the radial basis function, polynomial, exponential, and sigmoid kernel functions. Computational efficiency of the wave activation function was poor compared with the other kernel functions in the study. This research could provide a better understanding of the behavior of the kernel functions for support vector machine on the crude oil price dataset. The study has the potentials of triggering interested researchers to propose a novel methodology that can advance crude oil prediction accuracy.

Keywords: Radial basis function, Polynomial, Exponential, Sigmoid, Wave, Crude oil price.

1 Introduction

In machine learning, the use of kernels has attracted substantial attention of the research community. Such attention is attributed to the ability of the kernels in mapping data into a high dimensional feature space so that linear machine computational power can be increased. Kernel functions also allow the extension of linear hypotheses into nonlinear which can indirectly be achieved [1]. The success of the application of support vector machine in solving problems in classification, regression, clustering and density estimation critically depends on the appropriate selection, and use of a kernel function. A mechanism that can automatically choose

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_26, © Springer International Publishing Switzerland 2014

the correct kernel function for use in solving a particular problem is scarce in the literature. In a study by [2], effect of kernel functions on the sparsity of the solutions of relevance vector machine was conducted. However, the study do not statistically validated its results whereas the attention of the machine learning community has been drawn to that effect [3]. The performance of Gaussian and Maternal kernel functions were compared using root mean square error, and relevance vector's used to converge to the optimal solution. It was concluded that Gaussian is superior to Maternal. The simple difference observed could probably be caused by an estimation error. Therefore, statistical test with confidence bound is required to really measure the performance exhibited by the compared kernel functions [4].

The purpose of this research is to broaden the theoretic understanding of the effects of kernel functions for the support vector machine on crude oil price data. The crude oil price data are chosen as the subject of this research because of its global significance and research in this domain is still active. In this paper, we compared the performance of radial basis function, polynomial, wave, exponential and sigmoid kernel functions and the results were statistically validated.

Other sections of the paper are organized as follows: Section 2 presents the basic concept of the support vector machine algorithm. Section 3 described the methodology adopted in this research. Section 4 provides the experimental results and discussion, and lastly, concluding remarks and further research work is presented in section 5.

2 Support Vector Machine

Support vector machines are suitable algorithms for learning that are characterized by the control capacity of decision functions, and careful selection of appropriate kernel functions for use in solving problems [5]. In [6], Let (x_i, y_i), $i = 1, \ldots, l$ be a training set, where $x_i \in R^n$ and $y \in (+1,-1)^l$. Support vector machine requires the solutions of Equations (1) and (2)

$$\min_{w,b,\xi} \frac{1}{2} w^T w + C\sum_{i=1}^{l} \xi_i \tag{1}$$

Subject to

$$y_i \left(w^T Z_i + b\right) \geq 1 - \xi_i \, , \tag{2}$$

where

$$\xi_i \geq 0 \, , \; i = 1, \cdots, l \tag{3}$$

is mapped into high dimensional function space using function φ in Equation (4), i.e.

$$Z_i = \varphi(x_i) , \tag{4}$$

$$C > 0 ,$$

where *C* is the parameter of the error term, Equation (2) is solved by solving Equation (5) Subject to Equation (6) as follow:

$$\min \quad F(\alpha) = \frac{1}{2}\alpha^T Q_\alpha - \ell^T \alpha \tag{5}$$

$$0 \le \alpha_i \le C \ , \ i = 1, \cdots, l \ , \ y^T \alpha = 0 \ ,$$

where α and Q are vector of all 1's and lXl positive finite matrix, respectively. The i, j elements of Q is given by Equation (6) as follow:

$$Q_{ij} = y_i y_j K(x_i, x_j) \tag{6}$$

Where

$$K(x_i, x_j) \equiv \varphi(x_i)^T \varphi(x_j) \tag{7}$$

Equation (7) is referred to as kernel function, as such Equation (9) represents decision function as follow:

$$w = \sum\nolimits_{i=1}^{l} \alpha_i y_i \varphi(x_i) \tag{8}$$

$$\operatorname{sign}(w^T \varphi(x) + b) = \operatorname{sign}\left(\sum_{i=1}^{l} \alpha_i y_i k(x_i, x) + b \right) \tag{9}$$

Our interest in this research is on the Kernel functions in Equations 10–13 as they are among the most commonly used kernel functions in the machine learning literature.

Radial basis function

$$K(X_i, X_j) = \ell^{(-\gamma \|X_i - X_j\|^2)} \ , \ \gamma > 0 \ . \tag{10}$$

Polynomial

$$K(X_i, X_j) = (\gamma X_i{}' X_j + r)^d , \ \gamma > 0 \ . \tag{11}$$

Wave

$$K(X_i, X_j) = \frac{\gamma}{\left\|X_i - X_j\right\|} \sin\left(\frac{\left\|X_i - X_j\right\|}{\gamma}\right). \tag{12}$$

Sigmoid

$$K(X_i, X_j) = \tanh(\gamma X_i' X_j + r) \ . \tag{13}$$

Exponential

$$K(X_i, X_j) = \ell^{\left(-\frac{\left\|X_i - X_j\right\|}{\gamma}\right)}, \tag{14}$$

where, X_i is the inner product, X_j is the vector space, and r is the bias.

3 Experimental Setup

The experimental setup comprised of two major stages which includes the data collection, transformation, and preprocessing for improving the data standardization. In this manner, the support vector machine can efficiently perform computation without overflow. The second stage involves modeling of the support vector machine using the standardize data by trials with various kernel functions.

3.1 Dataset

Brent crude oil prices and West Texas Intermediate crude oil prices are generally considered as the international benchmark price by government, private businesses, and other key players around the globe. Prices in these oil markets have influence on other crude oil market prices as they are referring for the formulation of oil price [7]. In this paper, we chose the Brent crude oil price as our benchmark because two-third of the world makes reference to Brent crude oil price as pointed out in [8]. Monthly data from 1987 to 2012 are collected from US Department of Energy on the following variables: Organization of Petroleum Exporting Countries crude oil production (OPECCP), Organization for Economic Co – operation and Development crude oil ending stocks (OECDES), World crude oil production (WCOP), Non OPEC crude oil production (NOPECCP), US crude oil production, US crude oil imports (USCOI), Organization for Economic Co – operation and Development crude oil consumption (OECDCOC), US crude oil stocks at refineries (USCOSR), US gasoline ending stocks (USESTG), US crude oil production (USCOP), and US crude oil supplied (USCOS). In order to standardize the dataset and improve prediction accuracy, we normalized the dataset within the range of [–1,1].

3.2 Modeling the Support Vector Machine

The factors influencing Brent crude oil prices were subjected to sensitivity analysis in order to identify the most influential variable and handle it with care. The optimal parameters of the support vector machine are realized through several experimental trials using a small sample of the dataset. The larger dataset was partition into seven (7) ratios because the amount of the training data has a significant effect on the results produce by the support vector machine model. For consistent findings, each kernel function (radial basis function, polynomial, wave, exponential, and sigmoid) is used to perform seven (7) experiments with varying data partition ratio. The support vector machine is modelled to predict the prices of crude oil. Accuracy as well as computational time is observed during the experiments. Performance matrix adopted in this research is the widely use mean square error (MSE) and analysis of variance (ANOVA).

4 Results and Discussion

The results of the sensitivity analysis are depicted in Fig. 1. The factor with the highest influence is OPECCP as suggested by the experimental results. The sensitivity analysis is conducted in order to reveal the most significant factors, so that the users can handle such factors with great care and caution during the modeling process.

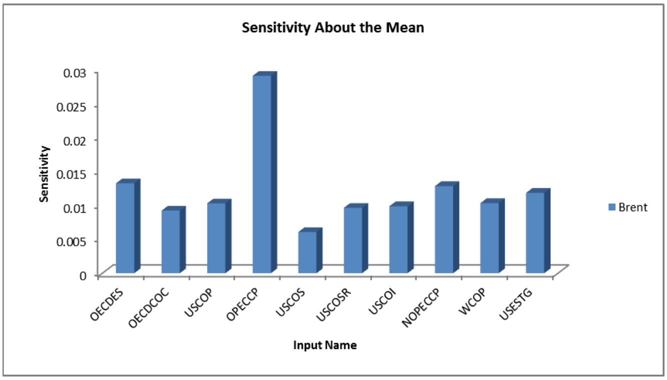


Fig. 1. Sensitivity analysis of the factors influencing Brent crude oil prices

In decreasing order, the five most influential factors are: OPECCP, OECDES, NOPECCP, USESTG, and USCOP. The OPECCP being the most influential factor is not surprising as the activity and action of OPEC typically affect the crude oil market significantly. The probable cause of this result could be due to the fact that the member countries of OPEC are among the top crude oil producers in the world.

Table 1. Optimal parameter of the support vector machine kernel functions

	Parameter of kernel			
Kernel Function	γ	**D**	**C**	**Support Vectors**
Radial basis	1	N/A	28	4321
Polynomial	4	9	98	3126
Wave	0.5	N/A	16	6342
Sigmoid	1.5	N/A	13	3434
Exponential	2.7	N/A	34	4561

Not Applicable (N/A)

Table 1 shows the results of the parameters realized after initial experimental trials in searching for the optimal parameters associated with each kernel function. Experimental results in Table 1 suggested that the wave has the largest number of support vectors. This likely occurred because the kernel function wave has the best accuracy among the compared kernel functions, and the higher the support vectors the more the accuracy as proved in [9].

Table 2. Comparing performance accuracy based on MSE

DP	RBF	Poly	Wave	Exp.	Sigmoid
60:40	0.092534	0.13451	0.0007534	0.2389	0.89421
65:35	0.781210	0.26722	0.004512	0.8934	0.34210
70:30	0.652900	0.56989	0.006590	0.6239	0.09452
75:25	0.231190	0.12941	0.0002345	0.4511	0.1298
80:20	0.216340	0.34768	0.009231	0.8211	0.3466
85:15	0.662340	0.17234	0.002333	0.4621	0.1209
90:10	0.038360	0.78231	0.0001758	0.1203	0.3543

Radial basis function (RBF), Polynomial (Poly), Exponential (Exp.), Data partition (DP)

From Table 2 the mean MSE of radial basis function, polynomial, wave, exponential, and sigmoid are 0.3821, .3433, .0034, 0.5158, and 0.3261 respectively. The wave is having the optimal performance among the compared kernel functions. To really determine the significant difference of the performance, we perform an ANOVA analysis. The results indicated that there is a significant difference [F (df=4, 30, $p<0.05$) = 3.993, Sig. = 0.10 at 95%] among the performance of the compared kernel functions. Therefore, we take additional steps to perform Post Hoc in order to determine which of the kernel functions has the mean significant. The Tukey results are reported in Table 3.

Table 3. The pairwise multiple comparisons with the performance accuracy (mean MSE) using Tukey's honesty test

(I) Group	(J) Group	Mean Difference (I-J)	Sig.	95% Confidence Interval Lower Bound	95% Confidence Interval Upper Bound
1.00	2.00	.03879	.998	-.3491	.4267
	3.00	.37872	.058	-.0092	.7666
	4.00	-.13370	.853	-.5216	.2542
	5.00	.05606	.993	-.3319	.4440
2.00	1.00	-.03879	.998	-.4267	.3491
	3.00	.33993	.108	-.0480	.7279
	4.00	-.17249	.699	-.5604	.2154
	5.00	.01728	1.000	-.3707	.4052
3.00	1.00	-.37872	.058	-.7666	.0092
	2.00	-.33993	.108	-.7279	.0480
	4.00	-.51242*	.005	-.9004	-.1245
	5.00	-.32266	.140	-.7106	.0653
4.00	1.00	.13370	.853	-.2542	.5216
	2.00	.17249	.699	-.2154	.5604
	3.00	.51242*	.005	.1245	.9004
	5.00	.18977	.621	-.1982	.5777
5.00	1.00	-.05606	.993	-.4440	.3319
	2.00	-.01728	1.000	-.4052	.3707
	3.00	.32266	.140	-.0653	.7106
	4.00	-.18977	.621	-.5777	.1982

It is clearly indicated in Table 3 that only Group 3 and 4 had significant mean difference at 0.05, whereas Group 1, 2, 5 had no significant mean difference at 0.05. Hence, is concluded that the performance accuracy of wave is significantly better than that for the radial basis function, polynomial, exponential, and sigmoid. The likely reason for these results could be attributed to the highly non-stationary nature of the wave which makes it fit well with the crude oil price data which is typically a non-stationary data.

The computational time for the support vector machine kernel functions to converge to the optimal solution is reported in Table 4.

Computation from Table 4 shows the mean time (seconds) require by radial basis function, polynomial, wave, exponential, and sigmoid to converge are 2.8571, 1.8571, 6.8571, 2.5714, and 2.1429, respectively. The wave is having the highest computational time among the compared kernel functions. The ANOVA analysis was performed. The output of the analysis indicated that there is a statistical significant difference [F (df=4, 30, $p<0.05$) = 17.939, Sig. = 0.00 at 95%] among the mean time of the compared kernel functions. Post Hoc was performed and the Tukey results are

Table 4. Comparing computational efficiency based on computational time in seconds

Data partition	RBF	Poly	Wave	Exp.	Sigmoid
60:40	2	2	6	3	3
65:35	1	2	8	2	2
70:30	3	3	5	2	3
75:25	2	2	7	3	2
80:20	4	1	6	1	1
85:15	1	1	9	3	2
90:10	7	2	7	4	2

Table 5. The pairwise multiple comparisons with the computational time using Tukey's honesty test

(I)	(J)	Mean Difference		95% Confidence Interval	
Group	Group	(I-J)	Sig.	Lower Bound	Upper Bound
1.00	2.00	1.00000	.594	-.9844	2.9844
	3.00	-4.00000*	.000	-5.9844	-2.0156
	4.00	.28571	.993	-1.6987	2.2701
	5.00	.71429	.833	-1.2701	2.6987
2.00	1.00	-1.00000	.594	-2.9844	.9844
	3.00	-5.00000*	.000	-6.9844	-3.0156
	4.00	-.71429	.833	-2.6987	1.2701
	5.00	-.28571	.993	-2.2701	1.6987
3.00	1.00	4.00000*	.000	2.0156	5.9844
	2.00	5.00000*	.000	3.0156	6.9844
	4.00	4.28571*	.000	2.3013	6.2701
	5.00	4.71429*	.000	2.7299	6.6987
4.00	1.00	-.28571	.993	-2.2701	1.6987
	2.00	.71429	.833	-1.2701	2.6987
	3.00	-4.28571*	.000	-6.2701	-2.3013
	5.00	.42857	.970	-1.5558	2.4130
5.00	1.00	-.71429	.833	-2.6987	1.2701
	2.00	.28571	.993	-1.6987	2.2701
	3.00	-4.71429*	.000	-6.6987	-2.7299
	4.00	-.42857	.970	-2.4130	1.5558

reported in Table 5. Despite the performance exhibited by the wave in this study, it was found that the performance of orthogonal kernel function reported in [10] is better than the wave kernel function in terms of both MSE and computational speed. The possible reason could be attributed to the capability of the orthogonal kernel function to capture the salient properties of the crude oil data. In addition, probably the wave function was not able to really utilize the OPECCP to maximum capacity during the prediction.

5 Conclusion

This paper presents a study that compared the performance of support vector machine's radial basis function, polynomial, Exponential, wave, and sigmoid kernel functions on the crude oil price dataset. The ANOVA results suggested that the performance exhibited by the wave kernel function is significantly better than that of radial basis function, polynomial, exponential, and sigmoid kernel functions. On the other hand, the wave had the poorest computational efficiency among the compared kernel functions. This research could provide a better understanding of the behavior of kernel functions on the crude oil price dataset, which might assist researchers in proposing novel approaches. We intend to further this research by repeating the experiments on several datasets.

Acknowledgments. This work is supported by University of Malaya High Impact Research Grant no vote UM.C/625/HIR/MOHE/SC/13/2 from Ministry of Higher Education Malaysia.

References

1. Genton, M.G.: Classes of Kernels for Machine Learning: A Statistics Perspective. J. Mach. Learn. Res. 2, 299–312 (2001)
2. Ben-Shimon, D., Shmilovici, A.: Kernels for the Relevance Vector Machine: An empirical Study. Adv. Web Intel. Data Min. 23, 253–263 (2006)
3. Demšar J.: Statistical Comparisons of Classifiers over Multiple Data Sets. J. Mach. Learn. Res. 7, 1–30 (2006)
4. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques (Data Management Systems). Morgan Kaufmann, San Mateo (2005)
5. Cristianini, N., Taylor, J.S.: An introduction to support vector machines and other kernel-based learning methods. Cambridge University Press, New York (2000)
6. Tipping, M.E.: Sparse Bayesian learning and the relevance vector machine. J. Mach. Learn. Res. 1, 211–244 (2001)
7. He, K., Yu, L., Lai, K.K.: Crude oil price analysis and forecasting using wavelet decomposed ensemble Model. Energy 46, 564–574 (2012)
8. Charles, A., Darne´, O.: The efficiency of the crude oil markets: Evidence from variance ratio tests. Energ. Policy 37, 4267–4272 (2009)
9. Demir, B., Ertürk, S.: Hyperspectral image classification using relevance vector machines. IEEE Geoscience Remotes 4(4), 586–590 (2007)
10. Chiroma, H., Abdul-Kareem, S., Abubakar, A., Akram, M., Zeki, A.M., Usman, M.J.: Orthogonal Wavelet Support Vector Machine for Predicting Crude Oil Prices. In: Herawan, T., Deris, M.M., Abawajy, J. (eds.) Proceedings of the First International Conference on Advanced Data and Information Engineering (DaEng 2013), vol. 285, pp. 193–201. Springer, Singapore (2014)

Modified Tournament Harmony Search for Unconstrained Optimisation Problems

Moh'd Khaled Shambour[1], Ahamad Tajudin Khader[1],
Ahmed A. Abusnaina[1], and Qusai Shambour[2]

[1] School of Computer Sciences, Universiti Sains Malaysia, 11800, Pulau Pinang, Malaysia
[2] Software Engineering Department, Al-Ahliyya Amman University, Amman, Jordan
myms11_com138@student.usm.my, tajudin@cs.usm.my,
abusnaina@ymail.com, q.shambour@ammanu.edu.jo

Abstract. Lately, Harmony Search algorithm (HSA) has attracted the attentions of researchers in operation research and artificial intelligence domain due to its capabilities of solving complex optimization problems in various fields. Different variants of HSA were proposed to overcome its weaknesses such as stagnation at local optima and slow convergence. The limitations of HSA have been mainly addressed in three aspects: studying the effect of HSA parameter settings, hybridizing it with other part of metaheuristic algorithms and the selection schemes that are used in selecting decision variables from harmony memory vectors. This paper focuses on improving the performance of HSA by introducing a new variant of HSA named Modified Tournament Harmony Search (MTHS) algorithm. The MTHS modifies the tournament selection scheme in order to improve the performance and efficiency of the classical HSA. Empirical results demonstrate the effectiveness of the proposed MTHS method and show its significance when compared with three benchmark variants of HSA.

Keywords: Harmony Search algorithm, Optimization problems, Metaheuristic algorithms.

1 Introduction

A Music-inspired Harmony Search Algorithm (HSA), introduced by Geem et.al. [1] as a population-based metaheuristic algorithm, has been applied fruitfully to a wide range of real-world optimization problems including: transport energy demand modelling, task assignment problem, job shop scheduling problem, knapsack problem, energy system dispatch, water distribution systems, web text mining, vehicle routing and timetabling problems [2-9].

HSA is characterized by a good functionality according to its ability of exploring the search space efficiently locking for global optima. However, it displays difficulties in performing local search for numerical applications [10]. Accordingly, several variations of HSA propose in the literature to cope with this problem [12-20].

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_27, © Springer International Publishing Switzerland 2014

The idea behind proposing a Modified Tournament Harmony Search (MTHS) algorithm lies in three main advantages: (i) increasing the process of convergence by selecting the best fit harmony vector according to its fitness value; (ii) maintaining the diversity of the solution vectors in HM in suitable level; and (iii) improving the exploration capacity which is raising the performance of HSA for finding the optima (or close to optima).

The rest of this paper is organized as follows: In Section 2, a brief overview of the HSA is presented. Section 3 demonstrates the MTHS algorithm, Section 4 presents the experimental setup and discuses the results. Finally, conclusions and directions for future study are provided in Section 5.

2 Harmony Search Algorithm

HSA mimics the process of musical players who are playing the pitches -from their musical instruments- searching for a good harmony as determined by an aesthetic standard. Each musician keeps the good pitches in his/her memory to replay it again hoping to increase the chance of generating a fantastic harmony in the next practice. Since that, each musician has three possible choices to improvise a pitch from his/her instrument: (i) improvise a memorized pitch; (ii) modify the memorized pitch; or (iii) improvise a new pitch from a possible range of pitches.

The analogy between improvisation process of musician in finding a new harmony and optimization is presented in Table 1. Geem et al. [11] formulized these options to three rules in improvising new harmony:

1. Harmony Memory Consideration (HMC): a decision value will be picked from stored solution in the Harmony Memory (HM) with probability of Harmony Memory Consideration Rate (HMCR).
2. Pitch Adjustment (PA): decides whether the picked decision value from the HM can be adjusted to neighboring values with probability of HMCR × Pitch Adjustment Rate (PAR).
3. Random Search (RS): a decision value picked randomly form defined range with probability of (1-HMCR).

Table 1. Musical terms vs. Optimization terms

Musical Terms	Optimization Terms
Musician	Decision Variable
Pitches of musical instruments	Values of variables
Improvising a pitch	Picking a decision variable
Musical harmony	Solution vector
Aesthetic standard	Objective function
Next practice	Next iteration
Fantastic harmony	(Near) global optimal

The optimization process of the HSA is as follows [1]:

1. Initialize the optimization problem and HSA parameters such as number of decision variables, values range, HMCR, PAR, Harmony Memory Size (HMS) and Number of Iteration (NI).
2. Initialize the harmony memory by generating solution vectors as the HMS.
3. Improvise a new harmony vector according to improvisation rules mentioned above.
4. Update the harmony memory by replacing the new solution vector with the respect to the worst solution vector in the HM (in terms of the objective function value).
5. Check for stopping criteria (NI is reached).

Classical HSA uses a random selection scheme to select decision variables from solution vectors in HM to build new harmony vector (new solution). In random selection, there is no bias involved in selecting decision variables (too low selection pressure) which leads to slow convergence and take longer to find the optimum. So that, a number of selection mechanism are applied [13,14,18,19,20] to enhance the process of selecting better harmony vectors from the HM which is leads to improve the quality of harmonies over succeeding generation.

3 The Proposed MTHS Method

In classical HSA, each harmony vector in HM represents a candidate solution to the optimisation problem. The MTHS algorithm uses the winner of the competitive harmonies from the tournament. The decision variables are selected randomly from different positions.

The MTHS algorithm (see Algorithm.1) starts by initializing the HM according to HMS with random numbers bounded to the problem domain. After that, a tournament with a random tournament size is applied to select the best harmony vector that will be used to select the decision variable randomly.

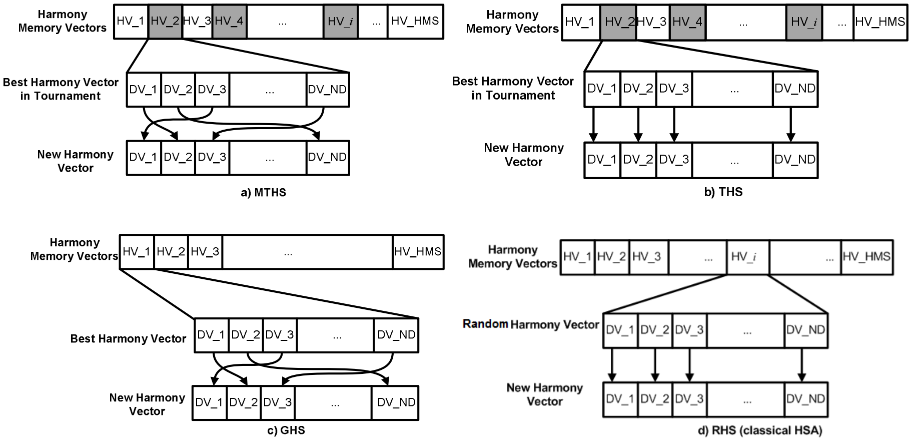


Fig. 1. Variants Harmony Search Algorithm. a)MTHS b)THS c)GHS d)RHS.

Fig. 1 demonstrates the variants of HAS (i.e. MTHS, THS [20], GHS [20] and RHS[1]), assuming that the harmony memory vectors are sorted according to their fitness (i.e. *f*(HV_1) < *f*(HV_2) ... < *f*(HV_HMS)) and the blocks with darker shades in the HM represent the harmony vectors competitors of the tournament selection mechanism.

Algorithm 1. MTHS Algorithm

```
Set HMCR, PAR, NI, HMS, BW,N.
$x_j^i = LB_i + (UB_i - LB_i) \times U(0, 1), \forall i = 1, 2,. . . N$ and $\forall j = 1, 2, . . . , HMS$
Evaluate $(f(\boldsymbol{x}_j)), \forall j = (1, 2, . . . , HMS)$
$itr = 0$
while ($itr \le$ NI) do
 $x' = \varphi$
 for $i = 1,..,$ N do
   if ($U(0, 1) \le$ HMCR) then //memory consideration
     $x^T =$ tournament (t, $x^1, x^2, ... x^{HMS}$) /* apply tournament selection with random
                     size $t \in \{1,2, ... HMS\}$ to select the best fit harmony from HM */
     $x_i' = x_k^T$, where $k \in \{1,2, ... N\}$
     if ($U(0, 1) \le$ PAR) then //pitch adjustment
       $\acute{x}_i = \acute{x}_i \pm U(0, 1) \times$ BW
     end if
   else //random selection
     $\acute{x}_i = LB_i + (UB_i - LB_i) \times U(0, 1)$
   end if
 end for
 if ($f(\acute{x}) < f(\boldsymbol{x}^{\text{worst}})$) then
   Include $\acute{x}$ to the HM.
   Exclude $\boldsymbol{x}^{\text{worst}}$ from HM.
 end if
 $itr = itr + 1$
end while
```

4 Empirical Experiments and Results

This section reports a number of experiments that demonstrate the effectiveness of the MTHS algorithm compared with a set of well-known benchmarking functions that are commonly used as variants of HSA in literature.

4.1 Benchmark Functions

Most of the benchmark functions used in this research, shown in Table 2, are multimodal functions except the step function which is unimodal and discontinuous. The multimodal functions have many local optimum points which exponentially increased with the dimension space [22]. The optimization algorithms have difficulties in finding the global optimum points due to the nonlinearity of benchmark functions among their variables [22]. Although the unimodal functions have single global optimum point, the computational methods suffer to find this point and it may give poor or slow convergence [23].

Table 2. Benchmark Functions

Fn. No.	Fn. Name	Expression	Range	Optimum
f_1	Rosenbrock	$f_1 = \sum_{i=0}^{d-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	$x_i \in$ [-30, 30]	0
f_2	Three-Hump-Camel	$f_2 = 2x_1^2 - 1.5x_1^4 + \frac{x_1^6}{6} + x_1x_2 + x_2^2$	$x_i \in$ [-5, 5] i=1, 2	0
f_3	Dixon Price	$f_3 = (x_1 - 1)^2 + \sum_{i=2}^{d} i(2x_i^2 - x_{i-1})^2$	$x_i \in$ [-10, 10]	0
f_4	Step	$f_4 = \sum_{i=1}^{d} (100 \lfloor x_i + 0.5 \rfloor)^2$	$x_i \in$ [-100, 100]	0
f_5	Schwefel's Problem 2.26	$f_5 = -\sum_{i=1}^{d} x_i \sin\left(\sqrt{\lvert x_i \rvert}\right)$	$x_i \in$ [-500, 500]	-12569.5
f_6	Griewank	$f_6 = \sum_{i=1}^{d} \frac{x_i^2}{4000} - \prod_{i=1}^{d} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$x_i \in$ [-600, 600]	0

4.2 Experimental Setup

The basic HSA parameters and coefficients are set in the initialization step of every algorithm (i.e. MTHS, THS, Global Harmony Search (GHS) and Random Harmony Search (RHS)). Same values are applied for all parameters where HMS=5, 10, 20, 50, 100, HMCR=0.96, PAR=0.6 and NI=5000 with dimensionality N=20. All methods are implemented in VBA 2010, under Windows 7 operating system. The experiments are performed on an Intel core 2 Quad CPU @2.27GHz with a memory of 4.00GB.

4.3 Results and Discussion

The MTHS algorithm is compared against THS [20], GHS [20], and RHS. The comparisons are conducted on the previously mentioned six benchmark functions. Each experimental session is repeated 30 times in which the best, mean and standard deviation of the objective function are reported.

The results (Appendix A, Table 3 and Table 4) show that the MTHS is a competitive alternative of other variants of HSA (i.e. GHS, THS and RHS). The MTHS achieved the best (lowest) mean values for f_1, f_2 and f_3. The mean values of f_4 and f_5 for both of MTHS and GHS are almost equivalent with minor differences, where both of them achieve the optimality (optimal results). The MTHS and GHS are alternately reach the best mean values for function f_6, this alternating behavior is a result of using different sizes of HM and the effect of the generated random numbers during the experimental sessions. Table-3 shows the results of f_1, f_2 and f_3, while Table-4 shows the results of f_4, f_5 and f_6.

In f_1, f_2 and f_3, the GHS achieves the best value out-of 30 times run but the mean is lower than MTHS. This is due to the ability of MTHS to avoid falling into the local optima, while others fail to escape from the local optima. The THS and RHS achieved the worst results and fail to reach the optimum or the near optimum solution, except that the RHS achieves the best mean for f_2 when the HMS was 20 and 100.

The best performance of MTHS is clearly observed when the search space is highly multimodal with many local optima (i.e. f_1,f_2 and f_3). Multimodal functions with many local minima considered to be one of the most difficult classes of problems for many algorithms [24].

Generating a new harmony from different harmony vectors -selected by the tournament selection method- gives the MTHS algorithm: (i) more chances to avoid getting stuck at local minima and (ii) enrich the diversity of HM solution vectors which make the solution space more explorative. On the other hand, GHS algorithm [20] has a high selective pressure on the global best solution which may lead to premature convergence problem since it selects the best harmony (best solution) among all other harmony vectors.

5 Conclusion

This paper proposes a new variant of the HSA called Modified Tournament Harmony Search (MTHS) algorithm which basically depends on modifying the tournament selection scheme in order to improve the performance and efficiency of the HSA.

The MTHS algorithm is compared against GHS, THS and RHS algorithms using six widely-used benchmark functions. The results show that the MTHS is better than or comparable with the others variants of HSA. The MTHS achieves the best mean value in three out of six functions and on par with GHS and RHS in two functions. Different parameters such as HMCR, PAR, HMS and NI affect the results. However, all of the compared algorithms (MTHS, GHS, THS and RHS) relies on the same parameter values.

The future work will (1) consider other complex multimodal benchmark problems; (2) examine the influence of the parameters setting (i.e. HMCR, PAR, HMS, and tournament size) on the optimization performance of the MTHS algorithm.

References

1. Geem, Z.W., Kim, J.H., Loganathan, G.V.: A new heuristic optimization algorithm: harmony search. J. Simulation 76(2), 60–68 (2001)
2. Ceylan, H., Ceylan, H.: Harmony search algorithm for transport energy demand modeling. In: Geem, Z.W. (ed.) Music-Inspired Harmony Search Algorithm. SCI, vol. 191, pp. 163–172. Springer, Heidelberg (2009)
3. Salman, A., Ahmad, I., Hanaa, A.R., Hamdan, S.: Solving the task assignment problem using Harmony Search algorithm. Evolving Systems, 1–17 (2012)
4. Yuan, Y., Xu, H., Yang, J.: A hybrid harmony search algorithm for the flexible job shop scheduling problem. Applied Soft Computing (2013)
5. Zou, D., Gao, L., Li, S., Wu, J.: Solving 0–1 knapsack problem by a novel global harmony search algorithm. Applied Soft Computing 11(2), 1556–1564 (2011)
6. Khazali, A.H., Kalantar, M.: Optimal reactive power dispatch based on harmony search algorithm. International Journal of Electrical Power & Energy Systems 33(3), 684–692 (2011)
7. Baek, C.W., Jun, H.D., Kim, J.H.: Development of a PDA model for water distribution systems using harmony search algorithm. KSCE Journal of Civil Engineering 14(4), 613–625 (2010)
8. Forsati, R., Mahdavi, M.: Web text mining using harmony search. In: Recent Advances in Harmony Search Algorithm. SCI, vol. 270, pp. 51–64. Springer, Heidelberg (2010)
9. Pichpibul, T., Kawtummachai, R.: Modified Harmony Search Algorithm for the Capacitated Vehicle Routing Problem. In: Proceedings of the International Multi Conference of Engineers and Computer Scientists, vol. 2 (2013)
10. Shambour, M.K.Y., Khader, A.T., Kheiri, A., Özcan, E.: A Two Stage Approach for High School Timetabling. In: Lee, M., Hirose, A., Hou, Z.-G., Kil, R.M. (eds.) ICONIP 2013. LNCS, vol. 8226, pp. 66–73. Springer, Heidelberg (2013)
11. Geem, Z.W., Choi, J.-Y.: Music composition using harmony search algorithm. In: Giacobini, M. (ed.) EvoWorkshops 2007. LNCS, vol. 4448, pp. 593–600. Springer, Heidelberg (2007)
12. Mahdavi, M., Fesanghary, M., Damangir, E.: An improved harmony search algorithm for solving optimization problems. Applied Mathematics and Computation 188(2), 1567–1579 (2007)
13. Omran, M.G., Mahdavi, M.: Global-best harmony search. Applied Mathematics and Computation 198(2), 643–656 (2008)
14. Pan, Q.K., Suganthan, P.N., Tasgetiren, M.F., Liang, J.J.: A self-adaptive global best harmony search algorithm for continuous optimization problems. Applied Mathematics and Computation 216(3), 830–848 (2010)
15. Doush, I.A.: Harmony search with multi-parent crossover for solving IEEE-CEC2011 competition problems. In: Huang, T., Zeng, Z., Li, C., Leung, C.S. (eds.) ICONIP 2012, Part IV. LNCS, vol. 7666, pp. 108–114. Springer, Heidelberg (2012)
16. Wang, C.M., Huang, Y.F.: Self-adaptive Harmony Search Algorithm for Optimization. Expert Syst. Appl. 37, 2826–2837 (2010)
17. Chakraborty, P., Roy, G.G., Das, S., Jain, D., Abraham, A.: An improved harmony search algorithm with differential mutation operator. Fundamenta Informaticae 95(4), 401–426 (2009)
18. Zou, D., Gao, L., Wu, J., Li, S.: Novel global harmony search algorithm for unconstrained problems. Neurocomputing 73(16), 3308–3318 (2009)

19. Pan, Q.K., Suganthan, P.N., Tasgetiren, M.F., Liang, J.J.: A self-adaptive global best harmony search algorithm for continuous optimization problems. Applied Mathematics and Computation 216(3), 830–848 (2010)
20. Al-Betar, M.A., Doush, I.A., Khader, A.T., Awadallah, M.A.: Novel selection schemes for harmony search. Applied Mathematics and Computation 218(10), 6095–6117 (2012)
21. Kattan, A., Abdullah, R.: A dynamic self-adaptive harmony search algorithm for continuous optimization problems. Applied Mathematics and Computation 219, 8542–8567 (2013)
22. Ortiz-Boyer, D., Hervás-Martínez, C., García-Pedrajas, N.: CIXL2: A Crossover Operator for Evolutionary Algorithms Based on Population Features. J. Artif. Intell. Res(JAIR) 24, 1–48 (2005)
23. Digalakis, J.G., Margaritis, K.G.: On benchmarking functions for genetic algorithms. International Journal of Computer Mathematics 77(4), 481–506 (2001)
24. Yang, X.S., Cui, Z., Xiao, R., Gandomi, A.H.: Swarm intelligence and bio-inspired computation: theory and applications. Elsevier (2013)

Appendix A

Table 3. Optimization Results of f_1, f_2 and f_3

Fn.	HMS	MTHS		GHS		THS		RHS	
		Best	Mean. ±(stdv)	Best	Mean. ±(stdv)	Best	Mean. ±(stdv)	Best	Mean. ±(stdv)
f_1	5	0.04955	**8.76202** (9.4)	0.04281	16.98566 (27.8)	101.24307	668.11672 (812)	131.626	685.442 (685)
	10	0.0448	**19.576** (33.317)	0.03949	14.794578 (37.49623)	73.22496	576.2576 (671.5)	203.2771	1258.86 (808)
	20	0.05428	**13.22** (21.051)	0.03353	14.45644 (30.3)	97.64846	440.35837 (320.38)	408.2333	1229.1 (623)
	50	0.03218	**5.5065** (8.4712)	0.04199	17.7305 (34.22)	152.42464	843.3584 (700.707)	365.4367	3007.71 (1762)
	100	0.05645	**6.7976** (8.970)	0.03962	9.34 93 (9. 449)	121.89065	982.14346 (743.390)	913.7323	6465.6 (3839.8)
f_2	5	1.19761E-11	**0.0175** (0.0659)	2.59991E-10	0.0543 (0.11646)	5.96573E-11	0.13936 (0.15153)	3.63867E-11	0.219 (0.13432)
	10	1.9243E-9	**0.00059** (0.00277)	1.19758E-12	0.01992 (0.0758)	4.64359E-11	0.1095 (0.14637)	1.68714E-11	0.13936 (0.15153)
	20	3.76295E-9	0.02297 (0.07641)	1.8776E-9	0.07695 (0.12957)	4.40685E-12	0.08959 (0.13919)	9.54037E-13	**0.02095** (0.0757)
	50	1.76579E-11	**0.00051** (0.00241)	2.00834E-10	0.01033 (0.05539)	1.98987E-12	0.1095 (0.14637)	2.9524E-11	0.03268 (0.09147)
	100	3.43994E-13	0.01007 (0.05496)	1.51445E-9	0.01746 (0.05774)	9.51047E-12	0.04977 (0.1132)	1.92032E-11	**4.75343E-8** (1.957E-7)
f_3	5	0.19036	**0.52719** (0.62381)	0.18411	0.95378 (2.30796)	0.12387	1.63048 (1.13774)	0.4943	2.42094 (1.57213)
	10	0.18531	**0.59099** (0.60104)	0.15616	0.59734 (0.80536)	0.71957	2.03152 (1.44477)	1.47684	6.63321 (4.08535)
	20	0.18693	**0.62399** (0.61255)	0.19384	18.25405 (96.96836)	0.13296	2.85729 (2.16138)	1.73037	9.80168 (7.84927)
	50	0.18539	**0.55691** (0.53011)	0.17896	0.61108 (0.74751)	0.71694	3.79227 (4.31947)	6.49191	23.86756 (12.11472)
	100	0.1909	**1.64509** (6.25659)	0.18486	12.00867 (62.31781)	0.81158	4.18107 (3.55148)	15.86887	51.34566 (27.89622)

Table 4. Optimization Results of f_4, f_5 and f_6

Fn.	HMS	MTTHS		GHS		THS		RHS	
		Best	Mean. ±(stdv)	Best	Mean.±(stdv)	Best	Mean. ±(stdv)	Best	Mean. ±(stdv)
f_4	5	0	**0** (0)	0	**0** (0)	9	39.13333 (23.65)	7	29.86667 (18)
	10	0	**0** (0)	0	**0** (0)	5	32.6 (18.28623)	11	33 (19.01542)
	20	0	**0** (0)	0	**0** (0)	10	36.13333 (17.155)	4	39.2666 (20.16)
	50	0	**0** (0)	0	**0** (0)	8	25.8666 (17.0106)	16	61.3 (27.132)
	100	0	**0** (0)	0	**0** (0)	5	25.0666 (23.1887)	39	99.8 (37.791)
f_5	5	-8379.657	**-8379.657** (0)	-8379.6577	**-8379.6577** (0)	-8305.74	-8225.89 (55.79)	-8305.28	-8230.23 (62)
	10	-8379.65773	-8379.63 (0.08)	-8379.6577	**-8379.6577** (0)	-8361.186	-8239.204 (73.6)	-8336.90	-8244.70 (58)
	20	-8379.65773	**-8379.65** (2E-4)	-8379.6577	**-8379.65** (1E-5)	-8340.423	-8234.1596 (85.75)	-8303.79	-8219.84 (57)
	50	-8379.65773	**-8379.6** (0.034)	-8379.6577	**-8379.65** (5E-6)	-8338.733	-8274.6068 (48.056)	-8312.594	-8214.12 (72.7)
	100	-8379.65773	-8379.5 (0.235)	-8379.6577	**-8379.65** (3E-6)	-8353.162	-8272.4907 (64.5)	-8328.11	-8213.14 (68)
f_6	5	1.000000023	**1** (2.2E-8)	1.00000002	**1** (3.2E-8)	1.10544	1.50553 (0.2578)	1.1466	1.4875 (0.22)
	10	1.000000028	1.00012 (4E-4)	1.00000002	**1.00002** (1E-4)	1.1967	1.40579 (0.1774)	1.13001	1.43 (0.188)
	20	1.000000028	1.00037 (0.00129)	1.00000001	**1** (9.4E-9)	1.10862	1.44184 (0.2072)	1.20578	1.543 (0.187)
	50	1.000000026	**1.000001** (5E-6)	1.00000001	1.00000113 (5E-6)	1.11414	1.3485 (0.15887)	1.29585	1.632 (0.245)
	100	1.000000029	1.00034 (8.4E-4)	1.00000002	**1.00000003** (8E-9)	1.15021	1.34896 (0.2071)	1.60589	2.0419 (0.35)

Multi-Objective Particle Swarm Optimization for Optimal Planning of Biodiesel Supply Chain in Malaysia

Maryam Valizadeh, S. Syafiie*, and I.S. Ahamad

Department of Chemical and Environmental Engineering, University Putra Malaysia,
43400 Serdang, Selangor, Malaysia
syafiie@upm.edu.my

Abstract. In this paper we develop a mathematical model for optimal planning of the biofuel supply chain. The model considers the optimal selection of feedstock while minimizing the total cost and social impact over the planning horizon. A multi-objective linear programming model (MOLP) is proposed to find the optimal solution. A multi-objective particle swarm optimization (MOPSO) method is applied to solve the mathematical model and it is compared with non-dominated sorting genetic algorithm (NSGA-II) . The model is used to evaluate the biodiesel production from palm oil and jatropha in Malaysia.

Keywords: Multi-objective optimization, MOLP, MOPSO, NSGA-II, Biofuel supply chain, Biodiesel, Palm oil, Jatropha.

1 Introduction

In recent years, rising oil price, the need for energy, concerns about its availability and climate change, caused looking for renewable energy to satisfy the demands [1]. Biofuel, derived from biological sources, is one of the renewable energy types. Recently, biofuels as an alternative for reducing oil dependence and environmental impact have attracted the attentions [2]. There are different elements in biofuel supply chain need to be considered for the performance of the entire supply chain. This paper presents the mathematical model for optimal planning of biodiesel supply chain based on existing facilities.

Modeling and planning of biofuel supply chain is an important issue for making decision in performance of the whole chain. Complexity of supply chain has led to look for new optimization methods. Recently, the heuristic optimization methods are being used instead of traditional methods because of their ability in finding an almost optimal solution [3]. Particle swarm optimization (PSO) is one of these heuristic methods. As the problem in hand to be solved is multi objectives, hence, multiple-objective optimization is considered.

There are several methods for solving multi-objective optimization problems. In most of the classical methods, a single objective is made by aggregating the

* Corresponding author.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_28, © Springer International Publishing Switzerland 2014

objectives into a single one. Another problem that accompanies traditional methods is that they produce only one solution in each run so it should be repeated several times to find the pareto optimal set [4]. PSO is the heuristic method based on the behavior of the birds within a flock which was proposed by Eberhart and Kennedy [5]. In PSO, each proposed solution called a particle which is a point in the search space of the optimization problem. Each particle flies in the multi-dimensional search space and its position will change depending on its experience and neighbors. In this method, the optimal solution is less dependent of the initial values and it is obtained after movement of all particles. PSO is easy to implement and few parameters needed to be adjusted. It is less sensitive to the objective function [6]. These advantages have led to the selection of PSO for this optimization problem.

A verity of models exist that present the optimal planning of biofuels supply chain. A linear programming (LP) and mixed-integer nonlinear programming (MINLP) models which account for the running of biomass-based energy and installation of processing plants at minimal operation cost in central Italy are described by Bruglieri et al. [7]. These models could handle single objective and are not applicable for multi-objective cases. The evaluation of the economic feasibility of bioenergy crops for producing bioethanol and biodiesel in Hawaii is considered by Tran et al. [8]. Some works have been conducted to consider the environmental performance of the biofuels supply chain in addition to the economic performance. Mele et al. [9] addressed the MILP model which considers the economic and environmental performance of sugar/ethanol production. However, the social aspect of the biofuel supply chain is not considered in this model. A multi-period MILP approach for the design and operation of biomass to liquid supply chains which considers the economic and environmental criteria is stated in the work of You et al.[10]. The model determines the optimal network, facility location, process technology, capital cost, level of inventory and planning of production and logistic decisions. To the best of our knowledge, optimal modeling of biofuel supply chain through the hybrid first and second generation biodiesel which takes into account economic and social objectives has not been considered to date.

2 Problem Definition

The structure of biofuel supply chain considered in this study consists of set of feedstock resources and pre-processing facilities, set of biorefineries and a set of demand zones. It is assumed that pre-processing facilities are located nearby the feedstock resources and also all of the facilities and biorefineries are installed. The goal of this study is development of the mathematical model for optimal planning of the biofuel supply chain based on available facilities which considers the minimization of the total cost and social impact through the entire planning horizon. The social impact is measured by the quantity of edible feedstock consumption.

3 Mathematical Model

A MOLP model is proposed to find the optimal solution for planning of biofuel supply chain. The model is developed to minimize the total cost and social impact over the planning horizon. The planning horizon is one year. Table 1 reveals the description of indices and input parameters and Table 2 shows the definitions of decision variables.

Table 1. Indices and parameters

Indices	
i	Feedstock types, $i = 1, ..., I$
l	Feedstock resources, $l = 1, ..., L$
w	Biorefineries, $w = 1, ..., W$
n	Demand zones, $n = 1, ..., N$
T	Time horizon
Parameters	
$Y_{i,l}$	Maximum availability of feedstock type i in resource l
η_i	Conversion factor of feedstock type i to pre-processed one
α_i	Conversion factor of pre-processed feedstock type i to biofuel
R_w	Maximum capacity of biorefinery w
D_n	Demand of demand zone n
$P_{i,l,w}$	Purchasing cost of pre-processed feedstck i from resource l
$T_{i,l,w}$	Transportation cost of pre-processd feedstock by truck
$D_{l,w}$	Distance between resource l and biorefinery w
$D\prime_{w,n}$	Distance between biorefinery w and demand zone n
$Ts_{i,l,w}$	Transportation cost of pre-processed feedstock by ship
$T\prime_{i,w,n}$	Transportation cost of biofuel by truck
$Ts\prime_{i,w,n}$	Transportation cost of biofuel by ship
$Pc_{i,w}$	Production cost of biofuel
β_i	Indicator equals 1 if feedstock is edible

3.1 Objective Functions

The first objective function represents the economic aspect of the biofuel supply chain and it measures the total cost. The total cost is the summation of feedstock purchasing cost (C_1), feedstock delivery cost (C_2), biofuel production cost (C_3) and biofuel transportation cost (C_4). These terms are given below:

$$C_1 = \sum_i \sum_l \sum_w X_{i,l,w} \cdot P_{i,l,w} \ . \tag{1}$$

$$C_2 = \sum_i \sum_l \sum_w (T_{i,l,w} \cdot D_{l,w} + Ts_{i,l,w}) \cdot X_{i,l,w} \ . \tag{2}$$

$$C_3 = \sum_i \sum_w X\prime_{i,w} \cdot Pc_{i,w} \ . \tag{3}$$

Table 2. Decision variables

Decision variables	
$Q_{i,l}$	Quantity of feedstock type i harvested from resource l
$C_{i,l}$	Quantity of pre-processed feedstock type i in pre-processing facility l
$X_{i,l,w}$	Quantity of pre-processed feedstock type i shipped from facility l to biorefinery w
$X\prime_{i,w}$	Quantity of biofuel produced from pre-processed feedstock type i in biorefinery w
$Q\prime_{i,w,n}$	Quantity of biofuel shipped from biorefinery w to demand zone n

$$C_4 = \sum_i \sum_w \sum_n (T\prime_{i,w,n} \cdot D\prime_{w,n} + Ts\prime_{i,w,n}) \cdot Q\prime_{i,w,n} \ . \tag{4}$$

Equation (1) represents the feedstock purchasing cost. Feedstock delivery cost is given in (2). Equation (3) shows the biofuel production cost. Biofuel transportation cost is expressed by (4). The delivery cost of feedstock and biofuel transportation cost are expressed in terms of delivery cost by ship and truck.

Therefore the total cost is:

$$\text{Total cost} = C_1 + C_2 + C_3 + C_4 \ . \tag{5}$$

The social objective of this study is minimization of edible feedstock consumption which is common between human food resources and feedstock used for biofuel production. Equation (6) represents the social objective function.

$$\text{Quantity of edible feedstock consumption} = \sum_i \sum_l Q_{i,l} \cdot \beta_i \ . \tag{6}$$

3.2 Constraints

The equality and inequality constraints are described as below:

Equation (7) represents that, due to the fact that it is not possible to harvest feedstock more than available amount, the total amount of feedstock collected from each resource cannot exceed its maximum yield.

$$Q_{i,l} \leq Y_{i,l} \quad \forall i \in I, l \in L \ . \tag{7}$$

It is assumed that pre-processing facilities are located nearby the fields and all harvested feedstock at each resource transferred to pre-processing facilities. The amount of pre-processed feedstock produced at each facility is given in below constraint.

$$C_{i,l} = Q_{i,l} \cdot \eta_i \quad \forall i \in I, l \in L \ . \tag{8}$$

Equation (9) shows that all pre-processed feedstock produced in pre-processing facilities, shipped from pre-processing facilities to biorefineries.

$$\sum_w X_{i,l,w} = C_{i,l} \quad \forall i \in I, l \in L \ . \tag{9}$$

Amount of biofuel produced from each type of feedstock equals the amount of pre-processed feedstock used for biofuel production multiplied by its relevant conversion factor. Equation (10) reveals to this constraint.

$$X\prime_{i,w} = \sum_{l} X_{i,l,w} \cdot \alpha_i \quad \forall i \in I, w \in W \ . \tag{10}$$

Total quantity of biofuel produced in each biorefinery should not exceed the maximum refinery capacity.

$$\sum_{i} X\prime_{i,w} \leq R_w \quad \forall w \in W \ . \tag{11}$$

Quantity of biofuel shipped from biorefineries to demand zones is given below:

$$\sum_{n} Q\prime_{i,w,n} = X\prime_{i,w} \quad \forall i \in I, w \in W \ . \tag{12}$$

Constraint (13) represents the demand satisfaction.

$$\sum_{i} \sum_{w} Q\prime_{i,w,n} \geq D_n \quad \forall n \in N \ . \tag{13}$$

Non-negativity constraints are given as:

$$Q_{i,l} \geq 0 \quad \forall i \in I, l \in L \ . \tag{14}$$

$$C_{i,l} \geq 0 \quad \forall i \in I, l \in L \ . \tag{15}$$

$$X_{i,l,w} \geq 0 \quad \forall i \in I, l \in L, w \in W \ . \tag{16}$$

$$X\prime_{i,w} \geq 0 \quad \forall i \in I, w \in W \ . \tag{17}$$

$$Q\prime_{i,w,n} \geq 0 \quad \forall i \in I, w \in W, n \in N \ . \tag{18}$$

4 Solution Strategy

There are several methods for solving multi-objective optimization problems. PSO is a heuristic method that simulate the social behavior of organisms within a group, such as birds, school of fish and so on. This method has advantages compared with other methods, such as optimal solution is less dependent of the initial values and it is obtained after movement of all particles. In addition, it is less sensitive and dependent of the objective function. The ease of implementation and simple setting are other advantages of this method [6].

The basic PSO cannot be used for multi-objective problems. There are several methods to extend PSO to handle multi-objective problems. The algorithm used in this paper is MOPSO proposed by Cagnina [11]. In this algorithm, a policy is defined to maintain the dominant solutions in iterations. Non-dominated solutions saved in external archive. The mutation operator was adapted to prevent premature convergence due to the local optima. Suppose that the search space has D dimension. Each particle is placed in the $X_i = [x_{i1}, ..., x_{iD}]$ with the velocity of $V_i = [v_{i1}, ..., v_{iD}]$. Each particle moves to the best position ever experienced ($pbest_i$). Velocity of each particle for each dimension and iteration is updated according to the *pbest*, position of the best particle (*gbest*) and certain velocity. Equation (19) represents the velocity.

$$v_i^{it} = \omega v_i^{it-1} + c_1 r_1 (pbest_i^{it-1} - x_i^{it-1}) + c_2 r_2 (gbest^{it-1} - x_i^{it-1}) \ . \tag{19}$$

where it and ω are number of iteration and inertia weight respectively. c_1 and c_2 are acceleration constants. r_1 and r_2 are random numbers between 0 and 1.

Position of each particle for each dimension is also updated in every iteration.

$$x_i^{it} = x_i^{it-1} + v_i^{it} \ . \tag{20}$$

The steps of algorithm used in this study is shown below:

1. Initialization of population and velocities.
2. Evaluation of particles based on objective functions.
3. Update the fitness vector.
4. Keeping non-dominated particles in external archive.
5. Selection of global best.
6. Update velocity and positions according to (19) and (20).
7. Apply mutation.
8. Evaluation of positions.
9. Update best position and fitness vector.
10. Update external archive.
11. Update global best.
12. Repetition of step 6 to 11 until the termination criteria met.

The global best is randomly selected from non-dominated particles in external archive. In this study, the process stopped if there is no significant improvement after specified number of iterations.

The MOPSO approach is compared with Non-dominanted sorting genetic algorithm (NSGA-II) method. NSGA-II is a common method for solving multi-objective optimization problems which is based on genetic algorithm. This algorithm calculates the optimal set using the non-dominated principle and crowding distance. Two performance metrics are used to evaluate the effectiveness of these methods. The first metric measures the number of pareto solutions (NOS). The higher number of pareto solutions shows that the method is more desirable. The second one is running time of algorithm.

5 Case Study

Biodiesel is a kind of biofuel derived from energy crops or fats. It is produced from the reaction of oil or fat and alcohol in presence of a catalyst. This process called transesterification. Studies show that more than 95% of biodiesel is made from edible oil [12, 13].

The model presented in this paper is used to evaluate the biodiesel production from palm oil and jatropha in Malaysia. Palm oil produced from oil palm fruit. Oil palm can be cultivated easily in humid areas such as Malaysia. Jatropha is a non edible crop that grows easily in all types of soils. The oil content of jatropha seed is about 30–40%. The potential area for jatropha plantation in Malaysia is 32.9 million acres [14–17].

5.1 Case Study Description

Malaysia has 13 states. Each state considered as a feedstock resource except Perlis due to the small area. The resources are not distributed uniformly in each state, so it is assumed that feedstock is available at the center of state. Various data sources were used for this case study. Area under oil palm plantation and oil palm fruit yield are based on data from Malaysian Palm Oil Board (MPOB). Jatropha harvested several times through a year in humid areas [18]. Potential area for plantation of jatropha in Malaysia is depicted in Table 3 [17].

Table 3. Potential area for plantation of jatropha

Region	Area (million acres)
Peninsular Malaysia	8.5
Sabah	10.4
Sarawak	14

Since jatropha grows in all types of soils even marginal lands, it is assumed that area under jatropha plantation in each state is proportional to the state area.

The average oil extraction rate (η_i) for palm oil and jatropha oil are 20.35% and 33% respectively[19].

According to the MPOB report presented in 2010, 12 biodiesel plants have been in operation and 4 plants have completed construction in Malaysia. These plants are located in 6 states. The total production capacity is 1.662 million ton/year [20]. Each of these states considered as a biorefinery location. We assume that biorefineries handle jatropha oil as well as palm oil.

Transportation is performed by diesel truck through west Malaysia. We choose ship for transportation between east and west Malaysia. [21] is used as a source for cost of transportation by truck for biodiesel. We note that all the costs are inflated using appropriate producer price index (PPI) and also costs in currencies except Malaysia currency are converted to Malaysian Ringgit (MYR). The ocean freights are obtained from Malaysian logistics buzz [22].

The variable cost of biodiesel production based on capacity of 100000 ton/year is obtained from the report addressing subsidies for biofuels in selected developing countries [23]. The fixed cost is not included in this study. Cost of purchasing crude palm oil and jatropha oil are obtained from MPOB and jatropha biodiesel website [24].

Transesterification process via an alkaline catalyst is applied to this study for production of biodiesel. The biodiesel yield in this process is about 99% [12].

Each state considered as a demand zone. The final demand for biodiesel in 2011 was 24 ktoe [25]. It is assumed that demand in 2012 is the same as 2011 and demand of each state is proportional to the population.

6 Implementation and Results

A MOLP model presented in this paper, was solved using MOPSO and NSGA-II methods. The model applied to the case study with combination of palm oil and jatropha oil as feedstock in Malaysia. The main decision variables are the quantities of raw materials to be harvested and selection of optimal set of feedstock in a way that minimize the total cost and edible feedstock consumption.

The MOPSO method applied to the case study has the population size of 200 particles. Empirical studies propose that the acceleration constants should be 2. The inertia weight is set to 0.4. MOPSO in this paper used mutation rate of 0.5. The process stopped if there is no significant improvement after 200 iterations.

The parameter settings for NSGA-II algorithm are initialized with the population size of 200, mutation rate of 0.5 and 200 iterations.

Values of performance metrics for both algorithms are depicted in Table 4.

Table 4. Values of performance metrics

Metric	MOPSO	NSGA-II
NOS	27	12
Time(Sec)	187.31	1379.52

Values of performance metrics represent that MOPSO algorithm has better result. It should be noted that high NOS metric and low running time is more desirable. The quantity of each type of feedstock is stated in Table 5.

Table 5. Quantity of feedstock to be harvested from each resource

Resouurce	l_1	l_2	l_3	l_4	l_5	l_6	l_7	l_8	l_9	l_{10}	l_{11}	l_{12}
Oil palm	3.9	204	195	114	180	108	5.3	25	81	1.2	170	167
Jatropha	199	328	247	264	156	160	178	336	245	263	179	358

in 1000 ton

7 Conclusion

The objective of this study was development of mathematical model for optimal planning of biodiesel supply chain. The main decision variable was selection of optimal quantity of feedstock to be harvested. A MOPSO method has been used to solve the optimization problem and compared with NSGA-II method. The model is applied to the case study in Malaysia. The results show that MOPSO method is more desirable and effective for planning of biodiesel supply chain.

Further development of this model could be incorporation of environmental criteria with social and economic objectives for optimal panning of biofuel supply chain.

Finally, in real world situations,there are some uncertainties in the biofuel supply chain which could impact the performance of the whole chain,so it should incorporate in future research.

Acknowlegment. This work is funded by ERGS project number ERGS/1-2012/ 5527093 supported by Ministry Of Higher Education Malaysia.

References

1. Aguilar, J.E.S., Campos, J.B.G., Ortega, J.M.P., Gonzalez, M.S., El-Halwagi, M.M.: Optimal Planning of a Biomass Conversion System Considering Economic and Environmental Aspects. Industrial and Engineering Chemistry Research 50, 8558–8570 (2011)
2. An, H., Wilhelm, W.E., Searcy, S.W.: Biofuel and Petroleum-based Fuel Supply Chain Research: A Literature Review. Biomass and Bioenergy 35, 3763–3774 (2011)
3. Silva, L.A.W., Coelho, L.S.: An Adaptive Particle Swarm Approach Applied to Optimization of a Simplified Supply Chain. In: 19th International Conference on Production Research
4. Coello, C.A.C.: A Comprehensive Survey of Evolutionary-Based Multiobjective Optimization Techniques. Knowledge and Information Systems 1, 269–308 (1999)
5. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. In: Proceeding of International Conference on Neural Networks, pp. 1942–1948. IEEE, Perth (1995)
6. Lee, K.Y., Park, J.B.: Application of Particle Swarm Optimization to Economic Dispatch Problem:Advantages and Disadvantages, pp. 188–192. IEEE (2006)
7. Bruglieri, M., Liberti, L.: Optimal Running and Planning of a Biomass-based Energy Production Process (2008)
8. Tran, N., Illukpitiya, P., Yanagida, J.F., Ogoshi, R.: Optimizing Biofuel Production:An Economic Analysis for Selected Biofuel Feedstock Production in Hawaii. Biomass and Bioenergy 35, 1756–1764 (2011)
9. Mele, F.D., Kostin, A.M., Gosalbez, G.G., Jimenez, L.: Multiobjective Model for More Sustainable Fuel Supply Chains. A Case Study of Sugar Cane Industry in Argentina. Industrial and Engineering Chemistry Research 50, 4939–4958 (2011)
10. You, F., Wang, B.: Life Cycle Optimization of Biomass-to-Liquids Supply Chains with Distributed-Centralized Processing Networks. Submitted Manuscript to Industrial & Engineering Chemistry Research (2011)

11. Cagnina, L., Esquivel, S., Coello, C.A.C.: A Particle Swarm Optimizer for Multi-Objective Optimization. JCS&T 5(4), 204–210 (2005)
12. Gui, M.M., Lee, K.T., Bhatia, S.: Feasibility of Edible Oil vs. Non-edible Oil vs. Waste Edible Oil as Biodiesel Feedstock. Energy 33, 1646–1653 (2008)
13. Agarwal, A.K., Das, L.M.: Biodiesel Development and Characterization for Use as a Fuel in Compression Ignition Engines. Journal of Engineering for Gas Turbines an Power 123, 440–447 (2001)
14. Lam, M.K., Tan, K.T., Lee, K.T., Mohamed, A.R.: Malaysian Palm Oil: Surviving the Food versus Fuel Dispute for a Sustainable Future. Renewable & Sustainable Energy Reviews 13, 1456–1464 (2009)
15. Divakara, B.N., Upadhyaya, H.D., Wani, S.P., Gowda, C.L.L.: Biology and Genetic Improvement of Jatropha Curcas L.: A Review. Applied Energy 87, 732–742 (2010)
16. Kalam, M.A., Ahamed, J.U., Masjuki, H.H.: Land Availability of Jatropha Production in Malaysia. Renewable and Sustainable Energy Reviews 16, 3999–4007 (2012)
17. Mofijur, M., Masjuki, H.H., Kalam, M.A., Hazrat, M.A., Liaquat, A.M., Shahabuddin, M., Varman, M.: Prospects of Biodiesel from Jatropha in Malaysia. Renewable and Sustainable Energy Reviews 16, 5007–5020 (2012)
18. Silip, J.J., Tambunan, A.H., Hambali, H., Sutrisno, S.M.: Lifecycle Duration and Maturity Heterogeneity of Jatropha Curcas Linn. Journal of Sustainable Development 3(2), 291–295 (2010)
19. Bionas Jatropha Biodiesel Project, `http://www.bionas.com.my`
20. MPOB, APOC: Palm Oil Development and Performance in Malaysia. Presentation to USITC Washington DC (2010)
21. Kim, J., Realff, M.J., Lee, J.H.: Optimal Design and Global Sensitivity Analysis of Biomass Supply Chain Networks for Biofuels under Uncertainty. Computers and Chemical Engineering 35, 1738–1751 (2011)
22. Malaysia Logistics Buzz, `http://www.malaysialogisticsbuzz.blogspot.com`
23. Lopez, G.P., Laan, T.: Biofuels-at What Cost? Government Support for Biodiesel in Malaysia. One of a series of reports addressing subsidies for biofuels in selected developing countries (2008)
24. Jatropha Biodiesel, `http://www.jatrophabiodiesel.org`
25. Malaysia Energy Information Hub, `http://www.meih.st.gov.my`

Nonlinear Dynamics as a Part of Soft Computing Systems: Novel Approach to Design of Data Mining Systems

Elena N. Benderskaya

St. Petersburg State Polytechnical University, Institute of Computing & Control,
194021, St. Petersburg, Politechnicheskaya 21, Russia
helen.bend@gmail.com

Abstract. In this article we will present the main steps of a new approach to design of Data Mining systems as well as its strengths and limitation. We will discuss how the structure of Soft Computing systems is formed through an incoming data in nonlinear dynamic systems. We will also give an example of the use of a chaotic dynamic system to solve a clustering problem under uncertainty (no a priori information about topology and number of clusters).

Keywords: data mining, soft computing, nonlinear dynamic system, clustering, chaotic dynamics, pattern recognition, Turing machine, metaheuristics.

1 Introduction

There are many methods to solve problems in various fields. Many methods are based on heuristics or on metaheuristics and it is not always clear which type of problem is best solved by a method, and what parameters will yield the best result for a given problem.

When solving any problem, the question arises of the best method to use in order to find a solution which satisfies the initial requirements. The classic approach is decomposition of the initial problem into sub problems and finding the best methods for each sub problem. Often, the researchers use only well-known methods or previously tested methods. In that case, there is a random component in the choice of methods of a large number of the many possible ones.

The difficulty of finding a suitable method, as well as developing a general method lies in the fact that the complexity of the method (and therefore the structure used) has to be adequate for solving the problem given. It would seem that the more complex the method, the wider the range of problems it can solve. However, simple problems, when solved by a complex method, often produce unsatisfactory results. Figuratively speaking, the additional degrees of freedom in the method, being unaffected by the input data, generate errors. This can be most clearly demonstrated on a neural network with an excessive number of elements for solving a simple problem. Instead of learning, with subsequent generalization the network does not produce patterns, but simply stores the input examples, and completely repeats the features of the training

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_29, © Springer International Publishing Switzerland 2014

examples, which may be related not to the features of the input space, but to the peculiarities of measurement and acquisition of data. As a result, incorrect results are obtained from the test data (or even worse, when the network is already in use).

Consider ways to ensure adequate structural complexity:

- principle of adjustment (development) of the structure (method) for a particular problem;
- synergic governance principles;
- principle of minimum description length.

In the first case the development of the neural network for a specific application is assumed, and thus the adequacy of the structure complexity and problem complexity is ensured [1]. To implement this method, developed by A. Galushkin, one must pass the a priori information to the primary and secondary optimization functions, which are then used to determine the adequacy of the structure.

A. Kolesnikov has developed a whole theory of synergistic control [2], with maximum use of the dynamics features of the object being controlled during development for ‘nonviolent’ control and maximum use of the object’s own dynamics to achieve a certain goal (subspace, trajectory or point) [2]. By using this approach, it is assured that the control system and control object are of adequate complexity. Similar to the first principle of adjusting to the problem is the principle of minimum description length, (proposed by A. Potapov) illustrated in detail for image recognition tasks [3]. To compare methods of problem-solving, a metric is created which corresponds to the length of the description of the method of solving the problem and thus the method with a minimum value of the metric is selected.

One of the main problems when developing AI systems is ensuring that the system is sophisticated enough for what needs to be solved. On the other hand such systems should be able to recognize both simple and sophisticated images, coming closer to the abilities of the biological counterparts. One of the possible ways of solving this dilemma is the development of AI systems which pick the methods and the parameters which are best for solving a certain problem. Search and application take place instead of developing new AI system in those cases when required AI systems exist for a subject area. These systems fall into three main categories:

- Multi-level automatic system with an expert at the highest level (advisory systems);
- A group of methods which solve problems by a majority rule;
- Universal methods (the results will be less accurate than specialized methods).

Let's look in more detail each of the options. The using of multi-level systems for making decisions supposes a preliminary assessment of the problem. Usually the higher level is automated, but not completely automatic and requires an expert. Advisory systems allow an increase in the number of possible methods to be considered in addition to those which the developer already knows when trying to find a solution. Also such systems allow us to take into account the knowledge accumulated about the features of each method, and based on task input data and the requirements for the solution, can give recommendations about the best method and optimum settings. However, the final decision is up to the expert developer.

Due to the complexity of a formal representation of the process of selecting the optimal method for solving a problem, a somewhat redundant but fairly effective approach can be offered – to create a system which would include most of the suitable methods, their use for solving the problem, and subsequent selection of the best solution by some quality criteria or majority rule [4]. For example, the systems of decision rule committees in the theory of pattern recognition and the formal algebra of events used on the set of these rules, designed by Yi. Zhuravlev [4].

The other design approach which was carried out by scientists from different fields is the creation of a fairly universal method of solving the problem which would be suitable for a large number of conditions for solving the problem, and would be insensitive to the deviation of the actual data from the data embedded a priori in the method. Such an approach generates methods that are universal, give results close to the optimal solution on average. In this case, the quality of the solution may be much lower than potentially achievable. This applies to the main indicators of quality, such as the probability of a correct solution, the accuracy of the solution, as well as to the secondary indicators - complexity, cost of memory and time consumed.

Analysis of the main existing approaches leads to the idea of a new approach that would combine all considered principles to ensure the adequacy of the system structure [1] for the complexity of the problem and to obtain general AI system. An approach will be proposed based on the assumption that the complexity of the system can be controlled by its response to dynamic changes in the input image directly. For this purpose, one can use a dynamic self-organizing system, which is sensitive to changes in input data and interpretation of the structure of the system and, accordingly, its complexity, must involve the concepts of the dynamic system complexity and the complexity of the attractor. By structural complexity in this case we mean not only the complexity of connections in the system itself, but also the complexity of the generated image in the phase space (attractor), which reflects the dynamics of the system and hence the complexity of the task.

2 Chaotic Dynamics – Next Stage of Soft Computing Complexity Level

Trend analysis of the mathematical apparatus of the static and dynamic point of view also leads to the idea of using a highly sensitive to changes in the input space nonlinear dynamic systems for the development of intelligent systems which includes in its dynamics all the possible problem solutions, simple as well as complex. And unlike the artificial construction of a universal approach, there is a universal system that organizes itself, adjusts to the solution [5].

Mathematical methods of nonlinear dynamics and chaos can be regarded as the next stage in the development of mathematical methods, since there is a tendency to shift from deterministic to statistical models with more complexity, to chaotic, which can be deterministic but due to nonlinearity and a large number of elements lead to complex and often unpredictable behavior.

Figure 1 is a schematic representation of development stages of the mathematical apparatus in terms of the complexity of methods, models and objects which can be described based on them. The development of mathematical methods and models from the point of view of a logic device (focus on static, the top part of Fig. 1) can be represented as follows. In the beginning there was classical logic which operates with clear numbers and precise sets. Largely this is why classical computational architectures require exact and specific input of the source data when performing calculations. In some complex and hard to formalize problems this is not possible.

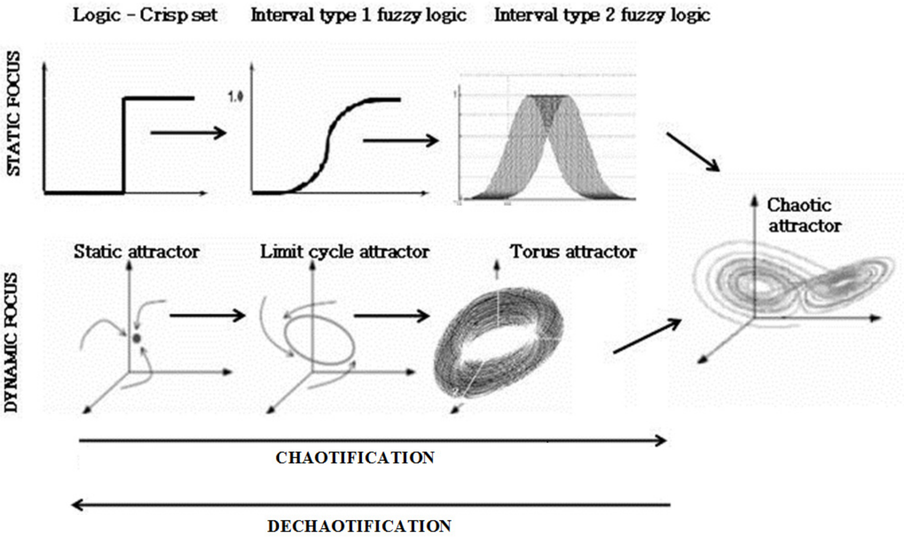


Fig. 1. Evolution of formal models: expanding the boundaries of models by introducing uncertainty (arrow from left to right) and constriction of the solution space to determine a particular solution (arrow from right to left)

A significant breakthrough in the field of information processing and overcoming linguistic uncertainty was the introduction of the concept of "fuzzy sets" and development of the theory of fuzzy logic. Now it is possible to perform operations simultaneously at a certain interval. The element on which the operations are performed is now an interval instead of a single point.

Further development of the theory of fuzzy sets and fuzzy logic is in some sense going via the extensive path: finding fuzzy sets of the second type, which are in reality "interval on an interval", increasing the dimension, etc. This, of course, enhances the capabilities of devices which deal with complexly organized and uncertain data, but, nevertheless, is not as effective as the transition from a number to interval.

One can observe the mathematical apparatus becoming more and more complex from the point of view of dynamic models when looking at the example of attractors attracting sets of dynamic systems as they become more complex (the bottom part of

Fig. 1). First, models of systems the dynamics of which converge to the set of individual points of attraction in the phase space (point attractor), then to the set of closed trajectories (attractor type: limit cycle, torus), and finally to a set of trajectories that define a location in the phase space in the form of an infinite number changing states (chaotic attractors). For static models, the next level of generalization, in order to extend the ability of making calculations simultaneously on a whole set of possible solutions, is also modeling with a chaotic attractor.

When looking at the trends in neural networks, we realize the necessity of using the capacity of chaotic dynamic systems for solving problems of AI and accomplishing related tasks (e.g. coding and information transfer). The functioning of the dynamic neural network with an irregular structure makes it possible to form a solution on the boundary of order-chaos, which corresponds to a variety of different structures of the output space, extremes of which are ordered dynamics (cycle) and turbulent dynamics (lack of structure in general).

This is the next step in the development of the neural network structure, as in this case, not only the weights of the network are adjusted, but a collective solution is found by a set of nonlinear elements of the same type, each one having unstable dynamics, but as a whole, under the influence of the input data, they form a stable dynamic system. It should be noted that another promising way of soft computing development is granular computing [6].

3 Structural Complexity of the System – Possible Way to Control

Control of chaos is often associated with the task of suppressing chaotic oscillations - the shift of the system to a stable periodic motion, or to a state of equilibrium. In a broad sense, it is the transformation of the chaotic behavior of the system into regular behavior or chaotic, but with different properties.

The challenges arising from the chaos control problem are much different from the traditional problems of automatic control [7]. Instead of classic control goals, such as bringing the trajectory of the system to a set point or to a given movement, soft goals are set to chaos control: creating modes with partially specified properties, qualitative change in the phase portrait of the system, synchronization of chaotic oscillations and others. Phenomenology of structure formation in nature inspired scientists to generate artificial systems with similar capabilities. One of the dominant ways to provide the collective dynamics of previously disordered elements is self-synchronization that occurs without any artificial enforcement [8].

Study of the dynamics of ensembles consisting of a large number of nonlinear elements, is one of the main trends in the theory of nonlinear oscillations and waves [9]. The main factor in the dynamics of ensembles of oscillating systems, which leads to an ordered-time behavior, is the synchronization of the ensemble elements. Numerous studies show that space-distributed random vibrating systems have many beneficial properties. In some of them self-synchronization occurs with specific parameters of the system. By self-synchronization we mean the process in which identical elements of the system, each of which is characterized by chaotic dynamics, is being initialized in various ways, over time, and starts to oscillate synchronously without outside influence.

In the presence of external influence on the nonlinear dynamic system, we get a response that reflects both the external conditions of the problem and the input signals which characterize the problem being solved. Instead of the usual representation of the original problem to be solved as a set of functions for subsequent use or for splitting the system into separate parts, in the synergetic approach the synthesis and study are performed on the system as a whole. Namely this, the occurrence of synchronization (collective behavior), allows living systems to adapt, learn, and extract information in real time to solve computationally complex problems (due to distributed information processing). Many elements with complex dynamics produce efficient computing [9-11].

A computing device that implements the proposed approach can be a set of asynchronous models of dynamic systems that interact with each other and combine properties such as being hybrid and asynchronous, having clusters (no rigid centralization and dynamic clustering of related models), and being stochastic [5, 10, 11]. O. Granichin developed a computational model for such a device that is based on the following set of basic parameters [10]:

- Set of computational primitives (dynamic models H_i with parameters from the set Q);
- Memory X - total space of states of all models;
- Feed S - dynamic graph with a finite bit string s of whether to include the models at certain nodes;
- Program G - The rules given by graph S are the rules (or goals) for “switches” of the tape and model parameters when the pair (x, q) appears at one of the “active” nodes in the switching set J;
- Cycle - the time interval between successive switches;
- Breakpoint set T.

One can speak of a generalization of a Turing machine [10] which can be represented as a chain of interrelated components $<A, H, Q, q, q0, X, x, x0, S, s, s0, J, G, T>$ where A is the set of models (computational primitives) , H - the evolution operator, Q - the set of states (parameter values), X – memory, S – the generic tape (graph), J – the set of switching, G - the program (goals), T – the breakpoint set. The main stages in the use of complex modes of operation of chaotic systems to solve practical problems can be represented by the following sequence.

The initial state is given and the goal is defined – to reach a certain state. It is assumed that the goal can be achieved by navigating through a trajectory that passes near one of the attractors. Then the system is started and the input signals corresponding to the task are given to it. After a transition process the system goes into an attractor. A search takes place for a trajectory which is accessible using a small perturbation of the system and is close enough to pass next to the desired point or sequence of points which corresponds to the desired state of the system. If such a path is not found, random input is fed into the system in order to jump to another attractor and so on until the goal is achieved.

The new method to design soft computing systems consists of 3 main phases. First phase is to provide the condition to self-organization by initializing the inputs and internal parameters of nonlinear system by means of input patterns and conditions of environment (using some metrics for direct transition). Second phase is self-organizing of nonlinear system by transition time and obtaining chaotic attractor. Third phase is interpretation of result structure by means of analysis of unstable dynamic and synchronization (using some metrics for reverse transition). The first phase corresponds to extension of solution space (chaotization, in analogy with fuzzification in fuzzy logics) and the third phase corresponds to constriction to real solution (de-chaotification, in analogy with defuzzification). Particular cases of the proposed general approach can be considered reservoir computing and chaotic neural network which discussed in more detail below.

4 Chaotic Neural Network – Example of Chaotic Attractor Approach in Data Mining Problem

In chaotic dynamics under the influence of external perturbation structures are produced, and it may initially include the entire set of possible options. Chaotic systems allow us to go to the next level of aggregation in the concept of process of computing and perform the calculations simultaneously on a whole set of possibilities, and this set will be shaped by external signals, thus providing an adequate complexity. In many ways, this is similar to the principles used in quantum computing, which contains the entire set of solutions until the answer is found.

Consider a relatively simple and clear example of the use of external images to form the structure of a system - the use of various metrics based on the input data for the calculation of the connection matrix in the chaotic neural network (CNN) [9, 12], capable of solving the problem of clustering only on the basis of input data without any additional a priory information about task.

A feature of this oscillatory neural network is chaotic dynamics of individual neurons' outputs, and mutual, independent of the initial conditions, self-clusterization. This allows for the use of CNN to solve problems with minimal prior clustering information about the objects to be sorted into clusters. CNN is a one-layer recurrent network in which the elements are connected to 'each other' without having a connection back to 'themselves':

$$y_i(t+1) = \frac{1}{C_i}\sum_{i \neq j}^{N} w_{ij} f(y_i(t)),\ t = 1...T, \tag{1}$$

$$f(y(t)) = 1 - 2y^2(t) \tag{2}$$

$$w_{ij} = \exp(-|x_i - x_j|^2 / 2a^2) \tag{3}$$

Where $C_i = \sum_{i \neq j} w_{ij},\ i, j = \overline{1, N}$ is a scaling constant, computed by the algorithm presented in [9, 12], w_{ij} is the connection strength (weight vector) between neurons *i* and *j*, *N* is the number of neurons, which is equal to the number of points in the input image, represented in the form of $\mathrm{X} = (x_1, x_2 \quad \ldots\ x_m)$, *m* is the dimension of the image space, and *T* is the simulation time. As shown in [9, 13], for nonlinear transformation $f(y(t))$ one can use any mapping that generates chaotic oscillations, however, a logistic mapping (2) is preferred.

The training of the CNN consists of assigning weight vectors, which are based on the ratio of the input image (3) and uniquely determine the field, which acts on all the neural networks. Because this field is not uniform, the analysis and resolution of the system of difference equations (1) is much more difficult. Study of the dynamics of ensembles of systems consisting of a large number of nonlinear elements is one of the main directions of development of the theory of nonlinear oscillations and waves. The main factor in the dynamics of ensembles of oscillating systems, which leads to an ordered space-time behavior of the ensemble, is the synchronization of the elements.

Analysis of the dynamics of different images for CNN (input structures, reflecting the impact of external environment on the system) with the same system parameters allows one to see the varying 'music' of vibrations at each of the clusters formed in the system. In Fig. 2, you can clearly distinguish ensembles of elements, the character of the output oscillations are very different, and allow one to talk of the existence of a system of self-generated clusters and the availability of fragmentary synchronization [12]. With this synchronization the instantaneous outputs of neurons belonging to the same cluster do not match either in amplitude or phase and do not have a fixed phase shift between any two sequences. By cluster fragmentation synchronization we mean synchronization in the sense that each cluster is characterized by a unique ‘melody’ of oscillations, encoded in the temporal sequence of output values of neurons. The proposed method for detection of cluster fragmentary synchronization is described in detail in [9, 12] and is based on an analysis of the relative remoteness of the instantaneous output values of each of the pairs of neurons in a varying time interval. Stable mutual synchronization of neurons within each cluster in terms of CNN corresponds to the macroscopic attractor. We receive oscillatory clusters which independent to initial conditions, though instant outputs of neurons differ greatly (Fig. 2). The complexity of mutual oscillations depends on the complexity of input image. Simple image comprised of points organized in compact groups located far from each other predetermines almost complete synchronization of oscillations within clusters (Fig 2a). But if the input image with less compact topology and less inter cluster distance than more complex synchronization take place (Fig. 2b). The system is stable in terms of mutual synchronous dynamics of outputs within time but not in terms of instant values of separate neurons.

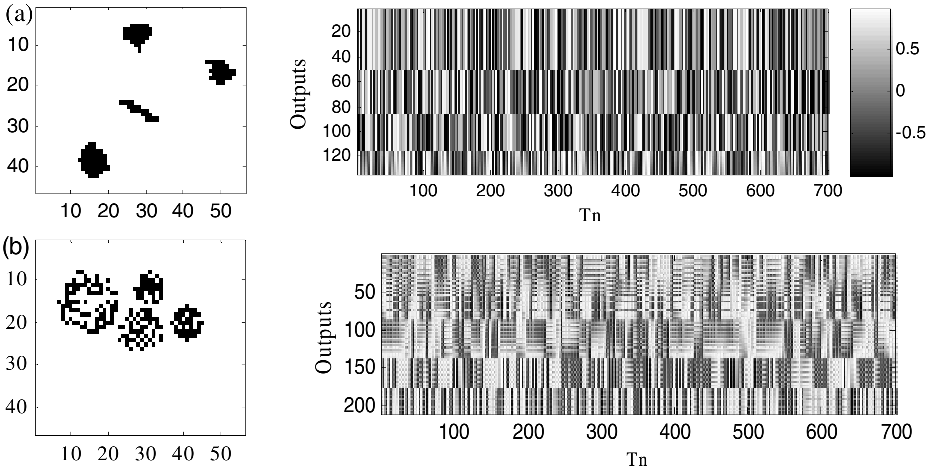


Fig. 2. Fragmentary synchronization for two different input images (one can see the "music" of oscillations of each cluster) (a) - simple clustering problem, (b) – more complex clustering problem

5 Conclusion

The need to address increasingly complex problems, and the opportunities that are provided when using synergistic principles of analysis and synthesis, lead to the idea that for the complex challenges that have manifested emergent properties, the more effective approach is a holistic analysis as a whole, without division. This is not a departure from functional decomposition, but a significant addition to it, since during fragmentation of the system we often lose the uniqueness associated with system patterns. One of the main goals of data mining is to obtain a new knowledge from the initial data and in some cases the result may be presented as a new structure or a new part of already obtained structure. It is good correlation with the structure formations in nonlinear systems with chaotic dynamics. A small influence to the input of the chaotic system leads to a new system state as well as a new conditions or a new data for data mining system leads to a new solution.

Thus, we propose a general approach to solving different data mining tasks - by reducing the original problem to a control problem, an optimization problem, or a problem of pattern recognition. This approach is similar to the neural network approach in the part, where problems of different types are reduced to the same type of problem and solvable by homogeneous network structures. In this approach, the complexity of the method (and the soft computing system to implement it) will be adequate to the complexity of the problem being solved just as it is in the formal synthesis theory of neural network structure through functions of primary and secondary optimization [1]. Above analysis demonstrates that nonlinear dynamical systems are a natural complement to existing and prospective soft computing systems. Proposed approach requires further development and refinement but the example of chaotic neural networks as a soft computing systems based on nonlinear dynamics indicates that the complex problem can be solved in unified form with high quality [9, 13].

References

1. Galushkin, A.I.: Neural Networks Theory. Springer, Heidelberg (2007)
2. Kolesnikov, A., Veselov, G., Monti, A., Ponci, F., Santi, E., Dougal, R.: Synergetic synthesis of dc-dc boost converter controllers: theory and experimental analysis. In: Conference Proceedings 17th Annual IEEE Applied Power Electronics Conference and Exposition, Dalas, TX, pp. 409–415 (2002)
3. Potapov, A.S.: Principle of representational minimum description length in image analysis and pattern recognition. J. Pattern Recognition and Image Analysis 22(1), 82–91 (2012)
4. Zhuravlev, Y.I.: An algebraic approach to recognition or classification problems. J. Pattern Recognition and Image Analysis 8(1), 59–100 (1998)
5. Benderskaya, E.N.: Nonlinear Trends in Modern Artificial Intelligence: A New Perspective. In: Beyond, A.I. (ed.) Beyond AI: Interdisciplinary Aspects of Artificial Intelligence. Topics in Intelligent Engineering and Informatics, vol. 4, pp. 113–124. Springer (2013)
6. Pedrycz, W.: Allocation of information granularity in optimization and decision-making models: Towards building the foundations of Granular Computing. European Journal of Operational Research 232(1), 137–145 (2014)
7. Andrievskii, B.R., Fradkov, A.L.: Control of chaos: method and applications. II Applications. J. Automation and Remote Control 65(4), 505–533 (2004)
8. Pikovsky, A., Rosenblum, M., Kurths, J.: Synchronization: A Universal Concept in Nonlinear Sciences (Cambridge Nonlinear Science Series). Cambridge University Press (2003)
9. Benderskaya, E.N., Zhukova, S.V.: Dynamic Data Mining: Synergy of Bio-Inspired Clustering Methods. In: Funatsu, K. (ed.) Knowledge-Oriented Applications in Data Mining, pp. 397–410. InTech publ. (2011)
10. Granichin, O.N., Vasil'ev, V.I.: Computational model based on evolutionary primitives. International Journal of Nanotechnology and Molecular Computation 2(2), 30–43 (2010)
11. Avros, R., Granichin, O., Shalymov, D., Volkovich, Z., Weber, G.-W.: Randomized algorithm of finding the true number of clusters based on Chebychev polynomial approximation. In: Holmes, D.E., Jain, L.C. (eds.) Data Mining: Found. & Intell. Paradigms. ISRL 23, vol. 1, pp. 131–155. Springer, Heidelberg (2012)
12. Benderskaya, E.N., Zhukova, S.V.: Fragmentary Synchronization in Chaotic Neural Network and Data Mining. In: Corchado, E., Wu, X., Oja, E., Herrero, Á., Baruque, B. (eds.) HAIS 2009. LNCS, vol. 5572, pp. 319–326. Springer, Heidelberg (2009)
13. Benderskaya, E.N., Zhukova, S.V.: Nonlinear approaches to automatic elicitation of distributed oscillatory clusters in adaptive self-organized system. In: Omatu, S., Paz Santana, J.F., González, S.R., Molina, J.M., Bernardos, A.M., Rodríguez, J.M.C. (eds.) Distributed Computing and Artificial Intelligence. AISC, vol. 151, pp. 733–741. Springer, Heidelberg (2012)

Soft Solution of Soft Set Theory for Recommendation in Decision Making

R.B. Fajriya Hakim[1], Eka Novita Sari[2], and Tutut Herawan[3]

[1] Department of Statistics,
Universitas Islam Indonesia,
Jalan Kaliurang KM 14, Yogyakarta, Indonesia
[2] AMCS Research Center, Yogyakarta, Indonesia
[3] Department of Information System,
University of Malaya,
50603 Pantai Valley, Kuala Lumpur, Malaysia
hakimf@fmipa.uii.ac.id, eka@amcs.co, tutut@um.edu.my

Abstract. Soft set theory is a new general mathematical method for dealing with uncertain data which proposed by Molodtsov in 1999 had been applied by researchers in decision making problems. However, most existing studies generated exact solution that should be soft solution because the determination of the initial problem only uses values or language approach. This paper shows the use of soft set theory as a generic mathematical tool to describe the objects in the form of information systems and evaluate using multidimensional scaling techniques to find the soft solution and recommendation for making a decision.

Keywords: Soft set theory, Decision making, Soft solution, Recommendation.

1 Introduction

Taking a decision is a matter that we encounter daily and most of the decisions we have taken are usually based on experience or a knowledge which we have seen before or we use the common things that can be used to assess the object. For example, when choosing furniture which will be bought, we usually describe as a classic, easy to clean, natural wood color, comfortable, and so on. It could be so when we have used all our knowledge to make choices, but we still felt unable to select it. Recommendations from others would be helpful in this kind of situation.

Molodtsov [1] has laid the foundation of a set that can collect different objects under consideration on the form of parameters they needed. This set which is called soft set is a set of parameters and a mapping function of those parameters to problems or objects under consideration. Molodtsov insisted that soft set could use any parameterization we prefer such as words and sentences, real numbers, functions and so on. This parameterization caused the multidimensional topological space. This model space needs not only metric space but also non metric space. Due to the basic notions of soft set that offers an approximate nature of the objects under consideration, the solution for someone's problem based on soft set should also be a

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_30, © Springer International Publishing Switzerland 2014

soft solution. Many research activities using soft set theory in decision making gave an exact solution, including the work of [2,3,4,5,6]. This paper shows a simple ranking evaluation applied for each object parameters in the soft set, mapping the parameters family of the objects using non metric multidimensional scaling and gives a soft solution that could be used as a recommendation for making decision.
The rest of this paper is organized as follow. Section 2 discusses rudimentary of soft set theory. Section 3 discusses related work on soft set and fuzzy-soft set based decision making. Section 4 discusses proposed soft set-based recommendation analysis, following by results and discussion. Finally, the conclusion of this work is described in Section 5.

2 Soft Set Theory

Molodtsov [1] first defined a soft set which is a family of objects whose definition depends on a set of parameter. Let U be an initial universe of objects, E be the set of adequate parameters in relation to objects in U. Adequate parameterization is desired to avoid some difficulties when using probability theory, fuzzy sets theory and interval mathematics which are in common used as mathematical tool for dealing with uncertainties. The definition of soft set is given as follows.

Definition 2.1. (See [1]). *A pair (F, E) is called a soft set over U if and only if F is a mapping of E into the set of all subsets of the set U.*

From definition, a soft set (F, E) over the universe U is a parameterized family that gives an approximate description of the objects in U. Let e any parameter in E, $e \in E$, the subset $F(e) \subseteq U$ may be considered as the set of e-approximate elements in the soft set (F, E).

As an illustration, the following is an example of soft set from Molodtsov [1].

Example 2.1. Let us consider a soft set (F, E) which describes the “attractiveness of houses” that Mr. X is considering to purchase.

U – is the set of houses under Mr. X consideration
E – is the set of parameters. Each parameter is a word or a sentence
E = {expensive, beatiful, wooden, cheap, in the green surroundings, modern, in good repair, in bad repair}

In this example, to define a soft set means to point out expensive houses, that shows which houses are expensive due to the dominating parameter is ‘expensive’ compared to other parameters that are possessed by the house, in the green surrounding houses, which shows houses that their surrounding are greener than other, and so on. Molodtsov [1] also stated that soft set theory has an opposite approach which is usually done in classical mathematics that should construct a mathematical model of an object and define the notion of the exact solution of this model. Soft set theory uses an approximate nature as an initial description of the objects under consideration and does not need to define an exact solution. In soft set theory, when someone is faced to the decision problems with many uncertainties, problem or object is determined by the ability of the person to explain various things related to that object. Related various things are referred to as the object parameter in the soft set. These parameters could be expressed by preference, knowledge, perception or a common word. Parameters attached to the objects are said to indispensable if the information involved to identifying a problem is sufficient to elucidate the objects. Setting the objects and their necessary information using words and sentences, real numbers, function, mappings, and etc. is a parameterization process that makes soft set theory applicable in practice.

Maji *et al.* [2] has extended Example 2.1 to decision making problem of choosing one of them based on its parameters. Some parameters are absolutely belonging to some houses and some parameters are absolutely not belonging by some houses. Their ideas have initiated many important applied and theoretical researches that have been achieved in soft set decision making problem. However, soft set theory has not been yet find out the right format to the solution of soft set theory due to many research using binary, fuzzy membership or interval valued for parameters of objects valuation which actually should be avoided as notified by Molodtsov [1].

3 Related Work

3.1 Soft Set-Based Decision Making

Maji *et al.* [2] applied the theory of soft set to solve a decision making problem that has been encountered by Mr. X. The best choice for Mr. X is only a house that has highest cumulative numbers based on the houses that have all the parameters. They also introduce the W-soft set or weighted soft set, but their method did not change the result. At this point, Maji *et al.* [2,7] assumed that the parameters as an attribute of an objects or object's features. This assumption is of course, need a process to transform the parameter to attribute/feature of objects or transform from soft set (F, E) over U to information system (U, AT) where AT is a nonempty set of attributes or features. Later, Herawan and Mat Deris [6] and Zou and Xiao [4] have proven that soft set could be transformed to binary information system.

Maji *et al* [2] also used rough set theory to reduce the parameters that have been hold to every object in the universe. Unfortunately, rather than optimize the worth of parameter as necessary information; they preferred to reduce the parameter. The information involved in the parameter will be loosed. Molodtsov has insisted the adequacy of parameterization to objects of universe rather than reducing the parameter that has been belonged to every object. Due to binary value of the entries, their decision result gives an exact solution that might contradicted to the philosophy of the initial Molodtsov's soft set that insisted the approximation to the result which is caused by soft information accepted in a parameterization family of soft set.

Chen *et al.* [3] and Kong *et al.* [5] also wanted to reduce the parameter of the objects, but Molodtsov already pointed out that the expansion of the set of parameters may be useful due to the expansion of parameters will give more detailed description of the objects. For example, Mr. X could add the parameter ‘distance to office’ for the attractiveness of houses. This parameter gives more detailed description of houses and may help him to re-decide which house to buy. Since the adequacies of parameter are crucial in soft set theory to describe the houses, reduction of parameters may bring out a set of indispensable parameter from a set of parameters. This reduction parameter expected to deduct valuable information from a set of indispensable parameter of soft set. For example, reducing the parameter ‘expensive’ and ‘cheap’ is allowed since Mr. X exactly knows that all houses has an actual same prices.

3.2 Fuzzy Soft Set-Based Decision Making

Roy and Maji [8] then used fuzzy set to evaluate the value of the parameter’s judgment for each object. This idea develops to the hybrid theory of fuzzy soft set. This fuzzy soft set also initiated by Yang *et al.* [9]. They prefer use the degree of membership than binary value. The difficulties are there must be an expert to determine the membership value that represents the matching number for each parameter of houses. It may become more difficult since the valuations of parameters of the objects are on the interval-valued fuzzy number [10,11]. An expert also should determine the lowest and the highest numbers as the value of the objects parameters. Molodtsov had been stated that this is the nature difficulties when dealing with fuzzy numbers and should be avoided. The idea to substitute the value of each object parameters using binary, fuzzy number or interval-valued fuzzy number may still be used as a reference because the results can be used as a benchmark for a person in making decision.

Several researchers could be categorized into two group that follows two main ideas i.e. treat a soft set as an attribute of information system [4,6], then using rough set (soft rough set) to handle the vagueness to make a decision [12], and the fuzzy soft set [10,11,13]. Both of them (Sections 2.1 and 2.2) gave techniques which produce best choice based on binary or fuzzy number rather than recommendation that may be little bit more satisfying Molodtsov’s philosophy [1].

This work will show a simple ranking evaluation applied for each object parameters in the soft set and mapping that parameters family to the subset of all set of the objects. This mapping is produced by using non metric multidimensional scaling as Nijkamp and Soffer [14] had introduced to soft multi-criteria decision models. The result is a soft solution or recommendation based on soft set which can be used as a suggestion for making decision.

4 Proposed Soft Set-Based Recommendation Analysis

4.1 Soft Solution

From Definition 2.1., a soft set (F, E) over the universe U is a parameterized family that gives an approximate description of the objects in U. Let e any parameter in E, $e \in E$, the subset $F(e) \subseteq U$ may be considered as the set of e-approximate elements in the soft set (F, E). As an illustration, let us consider example 1.1 given above. In this example, to define a soft set means to point out expensive houses, beautiful houses and so on. It is worth noting that the sets $F(e)$ may be arbitrary. Some of them may be empty, some may have nonempty intersection. That is, the solution of the soft set is a set which are a subset of object and a subset of parameters that shows the objects and its parameters.

Definition 4.1. (soft solution). *A pair (F', E') over U' is said to be a soft solution of soft set (F, E) over U if and only if*

i) $U' \subseteq U$

ii) $\{e_{|U'} \mid e \in E\} = E'$ *where* $e_{|U'}$ *is the restriction parameter of e to U'*

iii) F' is a mapping of E' into the set of all subsets of the set U'

We shall use the notion of restriction parameter of $e \in E'$ to U' in order to obtain the parameters which dominate an object compared to other parameters that may be possessed by those objects.

We are trying to approach the soft solution using information system theory which has been widely disclosed by Demri and Orlowska [15] that already has an established theoretical foundation of information system. Soft set theory is different from information system in which a problem or an object in the soft set is determined by the person dealing with the problem, then relies on the ability of him to be able to explain various things related to that object. Various things that might be related are referred to as the object parameter in the soft set. Meanwhile, an information system is a collection of objects and their properties. That is why, soft set described as a pair (F, E) over U with F is a mapping of E into the power set of U, and instead of (U, E) where U is the set of objects and E is the set of attributes as the structure of information system (OB, AT) where OB is the set of objects and AT is the set of attributes (properties). A formal information system (Demri and Orlowska, 2002) may be presented as a structure of the form (OB, AT, $(V_a)_{a\in AT}$, f), where OB is a non-empty set of objects, AT is a non-empty set of attributes, V_a is a non-empty set of values of the attribute a, and f is a total function $OB \times AT \rightarrow \cup_{a\in AT} P(V_a)$ such that for every $(x, a) \in OB \times AT$, $f(x, a) \subseteq V_a$. We often use (OB, AT) as a concise notation instead of a formal structure.

A soft set (F, E) over U might be considered as an information system (U, AT) such that $AT = \{F\}$ and the values of a mapping function of F are $e \in E$ which makes the existence of a sameness information about objects in U. It is a common thing to identify a wide range of matters (parameters) relating to the object and then create a collection of objects that possess this parameters. To compose this intuition, for a given soft set $S = (F, E)$ over U, we define a soft set formal context $S = (U, E, F)$ where U and E are non-empty sets whose elements are interpreted as objects and parameters (features), respectively, and $F \subseteq U \times E$ is a binary relation. If $x \in U$ and $e \in E$ and $(x, e) \in F$, then the object x is said to have the feature e. If U is finite then the relation F can be naturally represented as a matrix with entries $(c_{(x,e)})$ $x\in U$, $e\in E$ such that the rows and columns are labeled with objects and object parameters, respectively, and if $(x, e)\in F$, then $c_{(x,e)} = 1$, otherwise $c_{(x,e)} = 0$. In this concept, the soft set formal context provide the following mappings *ext*: $P(E)$ → $P(U)$, which shows extensional information for objects under consideration. This means an object parameters may be able to be expanded on someone views as the set of those objects that possess the parameters.

For all $X \subseteq U$ and $e \subseteq E$ we define $ext(E) \overset{def}{\Rightarrow} \{x \in U \mid (x,e) \in F$, for every $e \in E\}$; $ext(E)$ is referred to as the extent of E.

Lemma 4.1. *For all A, A_1, $A_2 \subseteq E$ if $A_1 \subseteq A_2$, then $ext(A_2) \subseteq ext(A_1)$.*

Each soft set formal context could be viewed as an information system. Given a soft set formal context $S = (U, E, F)$, we define the information system $(OB, AT, (V_a)_{a \in AT}, f)$ determined by S as follows :

i) $OB \overset{def}{\Rightarrow} U$ and $AT \overset{def}{\Rightarrow} E$

ii) For every $a \in AT$ and for every $x \in OB$, $f(x, a) \overset{def}{\Rightarrow} \{1\}$ if $(x, a) \in F$, otherwise $f(x, a) \overset{def}{\Rightarrow} \{0\}$

Any soft set formal context $S = (U, E, F)$ which has been viewed as an information system (OB, AT) could be represented as soft set information system $S = (U, E, F)$ that contains some information about relationships among parameter of the objects under consideration. This relation reflects a various forms of indistinguish-ability or 'sameness' of objects in terms of their parameters. Let $S = (U, E, F)$ be soft set information system. For every $A \subseteq E$ we define binary relations on U:

i) The indiscernibility relation $ind(A)$ is a relation such that for all x, $y \in U$, $(x, y) \in ind(A)$ if and only if for all $a \in A$, $a(x) = a(y)$.
ii) The similarity relation $sim(A)$ is a relation such that for all x, $y \in U$, $(x, y) \in ind(A)$ if and only if for all $a \in A$, $a(x) \cap a(y) \neq \theta$.

Intuitively, two objects are A-indiscernible whenever their sets of a-parameter determined by the parameter $a \in A$ are the same, while objects are A-similar whenever the objects share some parameters. In addition to having indistinguishability, we also show a formal distinguishability relation from a soft set information system,

i) The diversity relation $div(A)$ is a relation such that for all x, $y \in U$, $(x, y) \in div(A)$ if and only if for all $a \in A$, $a(x) \neq a(y)$.

Objects are A-diverse if all sets of their parameters determined by A are different. The information relations derived from soft set formal context S satisfy a property below,

Lemma 4.2. *For every soft set formal context $S = (U, E, F)$ and for every $A \subseteq E$, this assertion holds:*

i) $ind(A)$ is an equivalence relation.
ii) $sim(A)$ is reflexive and symmetric.

4.2 Primary Parameter in Soft Set

Let $S = (U, E, F)$ be a soft set information system and $A \subseteq E$. We say that parameter $a \in A$ is essential in A if and only if $ind(A) \neq ind(A\text{-}\{a\})$. It follows that if a is essential in A means, description of objects with respect to parameter from A is 'more' than the description based on $A - \{a\}$. The notion 'more' here is that the description based on indiscernibility of A-parameter provide a bigger partition and never lesser of the set of objects than the A-parameters objects without parameter a. The set A of parameters is independent if and only if every element of A is essential in A, otherwise A is dependent. This essential parameter property plays important roles in the soft solution. The absence of this parameter will cause a fundamental change to the outcome of soft solution. In the Molodtsov houses example, a parameter 'in the green surroundings' is essential, while parameter 'expensive' and 'cheap' or 'in good repair' and 'in bad repair' may could be selected one from each pair. The set of all parameters essential in A are referred to as the primary parameter of A in S:

$$\text{Primary}_S(A) \overset{def}{\Rightarrow} (a \in A \mid ind(A) \neq ind(A - \{a\})$$

Let S be soft set information system. By the discernibility matrix of S, the entries of the matrix $(c_{x,y})_{x,y \in U}$ where $c_{x,y} = c_{y,x}$ and $c_{x,x} = \emptyset$, $c_{x,y} = \{e \in E \mid (x,y) \in div(e)\}$. The columns and the rows of the matrix are labeled with objects whose entries are $c_{x,y}$.

Lemma 4.3. *Let $S = (U, E, F)$ be soft set information system and let $B \subseteq A \subseteq E$. Then the following assertions hold.*

i) $(x, y) \in ind(A)$ iff $c_{x,y} \cap A = \emptyset$;

ii) $ind(B) \subseteq ind(A)$ iff for all $x, y \in U$, ($c_{x,y} \cap A \neq \emptyset$ implies $c_{x,y} \cap B \neq \emptyset$);

iii) if $B \subseteq A$, then $ind(B) = ind(A)$ iff for all $x, y \in U$, $c_{x,y} \cap A \neq \emptyset$ implies $c_{x,y} \cap B \neq \emptyset$.

The above lemma enables us to find the primary of a set of parameter, namely we have the following theorem.

Theorem 4.1. *Let $S = (U, E, F)$ be soft set information system and let $A \subseteq E$. Then the following assertion holds:*

$$Primary_S(A) = \{a \in A \mid c_{x,y} \cap B = \{a\} \text{ for some } x, y \in U\}$$

This theorem says that parameter $a \in \text{Primary}_s(A)$ iff there are $x,y \in U$ such that a is the only parameter that allow us to make a distinction between x and y. In other words, the only division between x and y is provided by their a-parameter.

We have already present several reasons for soft set that might be treated as an information system. Bring the soft set to the structure of information system allows us to find a soft solution. In this work we use ranking value just look like simple fuzzy numbers to evaluate the parameter on soft set information system that may be done by all people rather set the membership value which may be done by an expert. Soft solution is expected come out using this way. This set of soft solution will be used in the recommendation analysis. The following is the recommendation analysis,

Definition 4.2. (Regions of recommendation). *Given a pair (F', E') over U' which is a soft solution of soft set (F, E) over U, a set $X \subseteq U'$ of objects and a set $A \subseteq E'$ of parameters, the regions of recommendation is defined as a A-parameters which are sufficient to classify all the elements of U' either as members or non-members of X. Since A-parameter might not be able to distinguish sufficiently between individual objects, it might be able to use cluster of objects rather than individual objects.*

Definition 4.3. (Recommendation analysis). *A recommendation analysis (R-analysis) of a soft set (F, E) over U which has a soft solution (F’, E’) over U’ where U’⊆U, E’⊆E and F’⊆F is a structure of the form R = (U’, F’, E’), where U’ is a non-empty set of objects under consideration, E’ is the restriction parameter of E and F’ is a mapping of E’ into the set of all subsets of the set U’.*

Adequate parameters used to draw the objects are a family of the set of parameter. The numbers of parameter that represents and draws the objects under consideration depends on someone’s view. One parameter could be regarded as minimal parameter if and only if this parameter could describe the objects under consideration, while many parameters could not be regarded as minimal parameter if and only if those parameters could not describe the objects under consideration at all.

Example 4.1. There are 6 houses under consideration (H1, H2, H3, H4, H5, H6) of Mr. X which will be bought, several parameter as in example 1.1 has been described to view the attractiveness of the houses. It is clear that such of his evaluation of parameter to each house is not very accurate; it is ‘soft’ because he used any qualitative and quantitative information or knowledge and perception he had, to judge each house. The simplest way to evaluate is by giving a rank or rate to each house based on parameter. Tabular representation would be useful to describe the response of Mr. X evaluation and might be checked using asterisks which describing the condition ‘more asterisks more meet parameters’.

Table 1. Evaluation of object parameters

U/E	Expensive	beautiful	Wooden	Cheap	In the green surroundings	Modern	In good repair	In bad repair
H1	***	**	***	*	**	**	***	*
H2	*	**	*	***	**	*	*	***
H3	***	**	*	*	***	**	***	*
H4	**	***	*	**	***	**	***	*
H5	*	*	*	***	**	**	*	***
H6	**	**	*	**	*	**	**	**

From Table 1, according to Mr. X, H1 is an as expensive as H3 even though the actual prices for both houses are not exactly same, H1 and H3 have highest prices among other houses and H4 and H6 are in the middle rate while H2 and H5 are the cheapest one. Some of parameters are vice versa, i.e., expensive and cheap, in good repair and in bad repair. A simple calculation to get the decision is collecting all the entries of each cell in the table based on the number of asterisks (Table 2).

Table 2. Cumulative evaluation

U	Total		
	***	**	*
H1	3	3	2
H2	3	2	3
H3	3	2	3
H4	3	3	2
H5	2	2	4
H6	0	6	2

From Table 2, H1 and H4, H2 and H3 have same value of attractiveness for several parameters but we could get difficulty to compare and knowing exactly what parameters they are. Table 2 could be regarded as a soft solution or recommendation to Mr. X for choosing one of them. So, the recommendation is {H1 and H4, H2 and H3, H6, H5}. This solution is very general, suppose Mr. X assigns several parameters which he thought as a priority parameter for purchasing house. Mr. X gives priority to parameter 'beautiful', 'cheap', 'modern' and 'in a good repair'. To get the solution, we should collect the entries of each cell in Table 2 based on the priority parameters and presented it in Table 3.

Table 3. Accumulation table based on priority parameters

U	Priority (P) Parameters			Normal Priority (N) Parameters		
	***	**	*	***	**	*
H1	1	2	1	2	1	1
H2	1	1	2	2	1	1
H3	1	2	1	2	0	2
H4	2	2	0	1	1	2
H5	1	1	2	1	1	2
H6	0	4	0	0	2	2

From Table 3, the recommendations for Mr. X are some choices, i.e., H4 due to H4 dominates others with two priority parameters have three and two asterisks, (H1, H3) or (H2, H5) because they have same values in the priority parameters, or H6 because H6 is the interesting one since it does not has any numbers in three asterisks but all priority parameters are in the middle valuation. The soft solution which is offered to Mr. X could be clustered into four groups {H4}, {H1 or H3}, {H2 or H5} or {H6}.

To better utilize information from the tables and provide added value to our recommendation for Mr. X, we will use multidimensional scaling techniques. Non-metric multidimensional scaling techniques are common techniques which based on ordinal or qualitative rankings of similarities data [16]. Using the software R [17] with *vegan* package and *metaMDS* procedure [18], we get the mapping of houses and its parameters ranking (Table 1) on Figure 1.

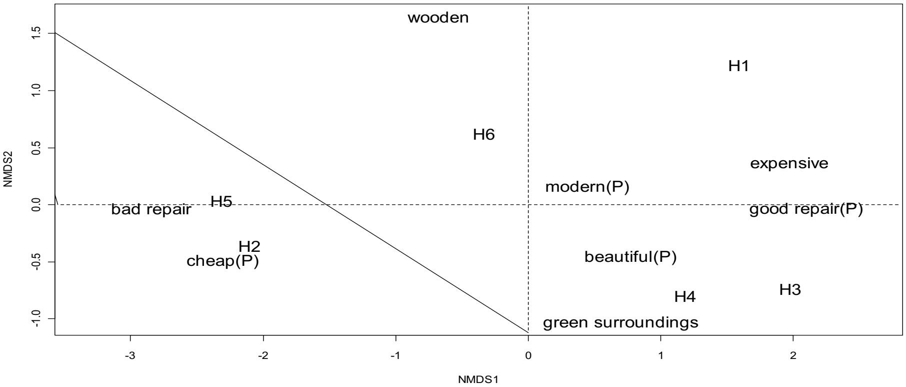


Fig. 1. Multidimensional scaling plot of houses and its parameters

From Figure 1 we get a map of houses and its parameters. We could see that, H3 and H4 are 'in the green surroundings', 'beautiful' and 'in a good repair' that two of its parameters are priority, H2 tends to 'cheap' (priority parameter), H5 closed to parameter 'bad repair', H1 be inclined to modern (priority parameter) and expensive, while H6 is in the middle evaluation. From this illustration we could consider a set as a solution of soft set as in the Definition 4.1. A set (*F'*, *E'*) over *U*' is a soft solution for soft set (*F*, *E*) over *U* if and only if *U*' is a subset of *U*, *E'* is the restriction parameter of *E*, *F'* is a mapping of *E'* into the set of all subsets of the set *U'*. The restrictrion parameter in this case is understood as a parameter which dominates an object. This may be taken in the following way, let say, *V* and *W* are sets of parameters, V, $W \subseteq E'$ where $V = \{ \upsilon_1, \upsilon_2, ... \}$ and $W = \{ \omega_1, \omega_2, ...\}$. We say that a set of parameters *V dominates W* on a set of all subsets of *U'* if and only if $\upsilon_i \geq \omega_i$ for every *i* and there exists an index *j* such that $\upsilon_j > \omega_j$.

Houses H3 and H4 closed to parameters 'green surroundings', 'beautiful' and 'good repair', but H4 is dominated by 'green surroundings' and 'beautiful' while H3 is dominated by 'good repair'. It could be explained, let V = {green surroundings, beautiful, good repair} and W = {good repair}, H4 is not dominated by *V* because there is v_3 = 'good repair' which is equal to w_1 = 'good repair' where $w_1 \in W$ that dominates H3. House H1 is inclined to parameter 'expensive' and 'modern', H6 is in the middle preferences. H2 is dominated by parameter 'cheap', while H5 is dominated by parameter 'bad repair'. Then the solution of soft set for Mr. X problem is

Soft solution (*F'*,*E'*) = {(green surroundings, beautiful) = H4, (good repair) =H3,
(expensive, modern) = H1, (cheap) = H2, (bad repair) = H5, ( ) = H6}

This set of soft solution is used as a basis for recommendations to Mr. X. So the recommendations for Mr. X is

{(H1, H2, H3, H4, H5, H6), (expensive, beautiful, wooden, cheap, in the green surroundings, modern, in good repair, in bad repair) | (green surroundings, beautiful) = H4, (good repair) = H3, (expensive, modern) = H1, (cheap) = H2, (bad repair) = H5, ( ) = H6}

H4 has two dominant parameters 'green surroundings' and 'beautiful' where 'beautiful' is priority parameter, H3 which has one dominant priority parameters 'in a good repair', H1 'expensive' and 'modern' where the 'modern' is a priority parameter or H2 which has dominant parameter 'cheap'. It is clear that these recommendations will help the decision makers to further solidify his choice and this set was not going to recommend one as the exact choice or the best choice. Recommendations like this are obviously very valuable because the decision makers can see the most important aspect of each object as well as increase knowledge for him to be steady in determining choice.

5 Conclusion and Future Work

In this work we have introduced an alternative way using simple ranking evaluation to parameters of soft sets. Molodtsov has emphasized the approximation or approach for soft set will produce a soft solution. Recommendation or soft solution describes the parameters which dominating for each choice. This example shows the strict dominant parameters that a set of all subsets of *U* have a set of parameters that have empty intersection with other set of parameters. Another kind of dominant parameter is a weak dominant parameter that allows a set of all subsets of *U* has a set of parameters that may have non empty intersection with other set of parameters. The main important part of soft set is adequate parameters of objects under consideration from someone that face with a choosing problem. The adequacy test is needed to test the adequacies of the parameters of soft set. Object parameters of soft set are called minimal if reducing one parameter will result the failure to give soft solution. One parameter is called adequate if and only if this parameter could describe the objects under consideration, while many parameters may be called inadequate if and only if those parameters could not describe the objects under consideration.

Many other decision making methods could be incorporated to this proposed recommendation analysis since the soft set theory already has the mathematical reason to be treated as an information system table. This paper has shown a useful study using a simple approximation to soft set theory in a decision making problem and still has many advancement in the use of real data or could be compared with other decision making methods.

Acknowledgments. This work is supported by University of Malaya High Impact Research Grant no vote UM.C/625/HIR/MOHE/SC/13/2 from Ministry of Higher Education Malaysia.

References

1. Molodtsov, D.: Soft Set Theory – First Results. Computers and Mathematics with Applications 37, 19–31 (1999)
2. Maji, P.K., Roy, A.R., Biswas, R.: An Application of Soft Sets in A Decision Making Problem. Computers and Mathematics with Applications 44, 1077–1083 (2002)
3. Chen, D., Tsang, E.C.C., Yeung, D.S., Wang, X.: The Parameterization Reduction of Soft Sets and its Applications. Computers and Mathematics with Applications 49, 757–763 (2005)
4. Zou, Y., Xiao, Z.: Data Analysis Approaches of Soft Sets under Incomplete Information. Knowledge Based System 21, 941–945 (2008)
5. Kong, Z., Gao, L., Wang, L., Li, S.: The normal parameter reduction of soft sets and itsalgorithm. Comput. Math. Appl. 56, 3029–3037 (2008)
6. Herawan, T., Mat Deris, M.: A soft set approach for association rules mining. Knowledge Based System 24, 186–195 (2011)
7. Maji, P.K., Roy, A.R., Biswas, R.: Soft Sets Theory. Computers and Mathematics with Applications 45, 555–562 (2003)

8. Roy, A.R., Maji, P.K.: A Fuzzy Soft Set Theoretic Approach to Decision Making Problems. Computational and Applied Mathematics 203, 412–418 (2007)
9. Yang, X.: Yu. D., Yang, J., and Wu, C., 2007, Generalization of soft set theory: From crisp to fuzzy case. In: Fuzzy Information and Engineering (ICFIE). ASC, vol. 40, pp. 345–354 (2007)
10. Feng, F., Jun, Y.B., Liu, X., Li, L.: An adjustable approach to fuzzy soft set based decision making. Journal of Computational and Applied Mathematics 234, 10–20 (2010)
11. Jiang, Y., Tang, Y., Chen, Q.: An adjustable approach to intuitionistic fuzzy soft sets based decision making. Applied Mathematical Modelling 35, 824–836 (2011)
12. Feng, F., Liu, X., Leoreanu-Fotea, V., Jun, Y.B.: Soft set and soft rough sets. Information Sciences 181, 1125–1137 (2011)
13. Jun, Y.B., Lee, K.J., Park, C.H.: Fuzzy soft sets theory applied to BCK/BCI-algebras. Computers and Mathematics with Applications 59, 3180–3192 (2010)
14. Nijkamp, P., Soffer, A.: Soft Multicriteria Decision Models for Urban Renewal Plans, Research memorandum no. 1979-5, Paper SistemiUrbani, Torino (1979)
15. Demri, S.P., Orlowska, E.S.: Incomplete Information: Structure, Inference, Complexity. Springer, Heidelberg (2002)
16. Kruskal, J.B.: Nonmetric multidimensional scaling: A numerical method. Psychometrika 29(1964), 115–129 (1964)
17. R Development Core Team, R: A language and environment for statistical computing, R Foundation for Statistical Computing, Vienna, Austria (2006), http://www.r-project.org/
18. Dixon, P., Palmer, M.W.: Vegan, a package of R function for community ecology. Journal of Vegetation Science 14, 927–930 (2003)

Two-Echelon Logistic Model Based on Game Theory with Fuzzy Variable

Pei Chun Lin[1] and Arbaiy Nureize[2]

[1] Graduate School of Information, Production and System, Waseda University,
2-7 Hibikino, Wakamatsu-ku, Kitakyushu, Fukuoka 808-0135, Japan
peichunpclin@gmail.com

[2] Faculty of Computer Science and Information Technology, University Tun Hussein Onn Malaysia, 86400 Batu Pahat, Johor, Malaysia
nureize@uthm.edu.my

Abstract. This paper applies Game Theory Based on Two–Echelon Logistic Models for Competitive behaviors in Logistics developed by Watada *et al*, which proposed the optimal decision method under two-echelon situation for logistic service providers. This study used three types of game theory; Cournot, Collusion, and Stackelberg to gain the optimizing strategies of exporters in each scenario. The aim of this paper is to realize optimal decision-making under competitiveness of these logistics service providers where they perform different game behaviors for achieving optimum solutions. Due to uncertain demand in the real world, fuzzy demands were applied for game theory in the two-echelon logistic model and compared results between fuzzy and non-fuzzy case. Numerical example is presented to clearly illustrate results by using fuzzy case and using crisp number. We obtain higher profits of both a shipper and forwarders when comparing the results yielded by non-fuzzy and fuzzy approaches.

Keywords: Fuzzy Variable, Game Theory, Decision Making, Two-Echelon Model, Logistics, Binomial Price Option Model.

1 Introduction

The echelon model is used in parts of supply chains, which are inventory, logistic, manufacturing, and so on. Two-echelon model or multi-echelon model is defined depending on situations [1]. This paper conducts an experimental research work on fuzzy variables in a game approach to evaluate the two-echelon model with forwarders and shippers that are competitively included in the logistics chain. This paper focuses on how determining their optimal decision for achieving optimum solutions in the competitiveness of the logistics service providers (LSPs) among the oligopolistic forwarders and the duopolistic shippers through performing different game behaviors. The main contribution of our paper is to apply fuzzy variables with the game theory approach for the logistic service. Game theory plays a pivotal role in optimizing the shipping performance while minimizing the overall shipping cost. The

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_31, © Springer International Publishing Switzerland 2014

shipper performs as a leader who provides shipping services to forwarders and directly to consignees (large manufacturer). The shippers need to determine the freight cost by negotiating with the downstream forwarders and the consumers (consignees). In addition to, this, a forwarder is a follower who consolidates small consignees (a small to medium manufacturer) to keep the cost down and the shipment quantity to shipper [2].

This paper analyzes the problem under different market condition, and also focuses on the quantity of the LSPs, which is commonly required to support the shipping decisions. In the shipping market, the shippers have to bear all the extra handling cost, which affects their profits. This paper also investigates the impact of the extra cost to the shippers in container shipping. To maximize the profit, the players do not concern the freight only but also the internal and external factor that impacts the cost. This paper addresses Cournot to explain freight and quantity decisions. It shows how firms compete on the amount of the output they produce. In Collusion competition, a cooperated action executes to gain mutual benefit among opponent firms, which are two or more players compete together. They communicate through other media and perform a particular strategy not truly saying. In Stackelberg, both competitors compete sequentially on the price or the quantity. The leader makes a move then the followers see this move and respond [3][4].

Furthermore, this paper proposes to apply fuzzy variables to unsteady demand in real situations. Fuzzy variables can be used to model various realistic situations, where uncertainty is not only in form of randomness but also in form of imprecision. Imprecision is described by means of fuzzy sets. Due to these, variables take fuzzy values instead of crisp ones. These situations involve the notion of a fuzzy variable. The concepts of fuzzy variables have membership function and possibility space to measure a fuzzy event [5-11]. The introduction of fuzziness leads to more probable two-echelon logistic models. In order to find the best strategies, which optimize the profits of the shipper and the forwarders, we recast the fuzzy demand two-echelon logistic problem as expected value models and obtain the analytical solution to each game behavior [12].

In the References [13-16], Lin, *et al.* has considered the central point and radius to define the defuzzification formula. We will consider this information in this paper. We also proposed the mean absolute percentage error of fuzzy demand (MAPRD_FD) in this paper. MAPRD_FD is the measure accuracy of a method for clarifying fuzzy number and crisp number. It will help us to choose a best profit in our problems. The structure of this paper is organized as theoretical explanation; numerical examples, conclusion and future work are explained.

2 Theoretical Definitions

In this section, we will give some theoretical and useful definitions and formulas we will use in this paper.

Let us introduce some definitions of triangular fuzzy numbers in the following.

Definition 1. Let $X = [a, b, c]$ be a triangular fuzzy number. Its membership function is written in the following form:

$$\mu_X(t) = \begin{cases} \dfrac{t-a}{b-a}; a \le t \le b \\ \dfrac{c-t}{c-b}; b \le t \le c \\ \quad 0;\ otherwise \end{cases}.$$

The expected value of triangular fuzzy number is defined as follows.

Definition 2. Expected Value of Fuzzy Data

Let $X = [a, b, c]$ be a triangular fuzzy numbers on a probability space $(\Omega, \mathrm{F}, \mathrm{P})$, where is a sample space, F is a sigma-algebra, and P is a probability measure. Then, the expected value of triangular fuzzy number X is defined as follows.

$$E(A) \equiv \big(E(a), E(b), E(c)\big).$$

To defuzzify the fuzzy data, Lin, *et al.* [16] have defined a weight function by using the central point and radius. We will adopt the information to define defuzzification formula in this paper. We give the definition of central point and radius in the following. The complete proof has given in Reference [17].

Definition 3. Let $X = [a, b, c]$ be a triangular fuzzy number, then the central point and radius are written as $o = \frac{a+b+c}{3}$ and $l = \frac{c-a}{4}$, respectively. Moreover, the expected value of central point and radius are $\mathrm{E}(o) = \mathrm{E}\left(\frac{a+b+c}{3}\right) = \frac{\mathrm{E}(a)+\mathrm{E}(b)+\mathrm{E}(c)}{3}$ and $\mathrm{E}(l) = \mathrm{E}\left(\frac{c-a}{4}\right) = \frac{\mathrm{E}(c)-\mathrm{E}(a)}{4}$, respectively. [17]

3 Two-Echelon Model for Fuzzy Numbers

3.1 Model Assumptions

Four cases are considered to analyze two-echelon model in order to investigate three competitive situations: Collusion, Cournot, and Stackelberg.

Case1: The total requirement to the forwarders is more than the direct requirement to the shippers directly from the consignees, but the forwarders cannot respond to the entire requirement from the consignees. The forwarders get the requirements from various manufacturers by sell at low freight service; the shippers obtain profit from selling shipments to the forwarders. The shippers require paying their additional service costs from forwarders' consignees.

Case2: The total requirement to the forwarders is more than the direct requirement from the consignees to the shippers. Therefore, the forwarders can respond the services (Long-term). This case is almost similar to Case1 except the demand of the forwarders continuously increases by a long-term contract with the consignees. The shippers' profit and additional costs are the same as Case1.

Case3: The total requirement to the forwarders is less than the direct requirement from the consignees to the shippers. It happens in the beginning of peak shipping season, a shipper sets freight to competitive price and obtain profit from direct consignees. There is a small demand request from forwarders, so the additional service cost is reduced.

Case4: The total requirement to the forwarders is still less than the direct requirement from the consignees to the shippers (Long-term). It happens during a peak season, a shipper trends to move to a long-term contract with direct consignees and get much larger shipment. Shippers obtain great profit by themselves; therefore a small additional cost is necessary to pay as it as in the previous case.

Constraints are listed below:

1. The total shipment consists of the amount of direct services from shippers to consignees and a service provide to forwarders.

2. Shippers compete not only between their competitors but also downstream LSPs to gain more shipping quantity.

3. There are two forwarders i, j in this study where forwarder i provides better service with more reasonable price than forwarder j. On the other hand, forwarder j provides lower freight price to attract more consignees without providing an additional service.

4. Sometime a shipper is requested to pay extra additional cost for forwarder j's service due to bad service.

5. All additional costs that are paid by a shipper are included additional operation cost (β), extra container inventory cost (α), and extra handling cost (θ).

6. This model illustrates a Stackelberg structure between Shipper (leader) and Forwarders (followers).

We assume that shippers and forwarders face the following toward-sloping demand function:

Shipper quantity: $$Q_s = (D_s - aP_s + bP_s^{'}) \tag{1}$$

Forwarder i quantity: $$Q_i = (D_i - cP_i + dP_j) \tag{2}$$

Forwarder j quantity: $$Q_j = (D_j - cP_j + dP_i) \tag{3}$$

where parameters of the shippers and forwarders $D_s > 0, a > 0$, $a < b < 0$, $D_i > 0, D_j > 0, c > 0$, and $c < d < 0$. Moreover, D_s, D_i and D_j are fuzzy demands.

3.2 Two-Echelon Model for Fuzzy Numbers

In this section, we mainly consider the situations where the duopolistic retailers implement the three scenarios below:

1) Finding their Cournot solution, each forwarder independently sets his retail price and orders quantity by assuming his rival's freight price as a parameter.

2) Acting in collusion, both forwarders are willing to design their sale prices jointly to maximize the total profit of the downstream retail market.

3) Playing a Stackelberg game, forwarder i is a leader, and forwarder j is a follower.

So, we can determine shippers' profit function in each case of role:

Case 1: $$\pi_s^* = (P_s + \delta P_s + \theta P_s)Q - \delta C_s Q_s - P_i^* Q_i - P_j^* Q_j - \beta C_s \alpha Q_j \quad (4)$$

Case 2: $$\pi_s^* = (P_s + \delta P_s + \theta P_s)Q - C_s Q_s - P_i^* Q_i - P_j^* Q_j - \beta C_s \alpha Q_j \quad (5)$$

Case 3: $$\pi_s^* = (P_s + \delta P_s + \theta P_s)Q - \delta C_s Q_s - P_i^* Q_i - P_j^* Q_j - \beta C_s \alpha Q_j \quad (6)$$

Case 4: $$\pi_s^* = (P_s + \delta P_s + \theta P_s)Q - P_i^* Q_i - P_j^* Q_j - \beta C_s \alpha Q_j \quad (7)$$

When the two-echelon model for two forwarders pursues, three behaviors are shown as follows. The proof was shown in the paper of Reference [5].

1) *Cournot solution*:

$$P_i^* = [2cD_i + dD_j + cdC_j + 2c^2C_i)]/(4c^2 - d^2) \quad (8)$$

$$P_j^* = [2cD_j + dD_i + cdC_i + 2c^2C_j)]/(4c^2 - d^2) \quad (9)$$

$$Q_i^* = \frac{2c^2}{4c^2-d^2}D_j + \frac{cd}{4c^2-d^2}D_i + \frac{cd-2c^3}{4c^2-d^2}c_i + \frac{c^2d}{4c^2-d^2}c_j \quad (10)$$

$$Q_j^* = \frac{2c^2}{4c^2-d^2}D_i + \frac{cd}{4c^2-d^2}D_j + \frac{cd-2c^3}{4c^2-d^2}c_j + \frac{c^2d}{4c^2-d^2}c_i \quad (11)$$

2) *Collusion solution*:

$$P_i^* = \frac{2c}{3c^2+2cd-d^2}D_i + \frac{c-d}{3c^2+2cd-d^2}D_j + \frac{4c^3-c^2+cd}{3c^2+2cd-d^2}(C_i + C_j) \quad (12)$$

$$P_j^* = \frac{2c}{3c^2+2cd-d^2}D_j + \frac{c-d}{3c^2+2cd-d^2}D_i + \frac{4c^3-c^2+cd}{3c^2+2cd-d^2}(C_i + C_j) \quad (13)$$

$$Q_i^* = \frac{c^2+cd}{3c^2+2cd-d^2}D_i + \frac{c^2+cd}{3c^2+2cd-d^2}D_j + \frac{4c^3d-2c^2d+c^3+cd^2-4c^4}{3c^2+2cd-d^2}(C_i + C_j) \quad (14)$$

$$Q_j^* = \frac{c^2+cd}{3c^2+2cd-d^2}D_j + \frac{c^2+cd}{3c^2+2cd-d^2}D_i + \frac{4c^3d-2c^2d+c^3+cd^2-4c^4}{3c^2+2cd-d^2}(C_i + C_j) \quad (15)$$

3) *Stackelberg solution*:

$$P_i^* = \frac{c}{2c^2-d^2}D_i + \frac{d}{4c^2-2d^2}D_j + \frac{1}{2}C_i + \frac{cd}{4c^2-2d^2}C_j \quad (16)$$

$$Q_i^* = \frac{1}{2}D_i + \frac{d}{4c}D_j + \frac{d^2-2c^2}{4c}C_i + \frac{d}{2}C_j \quad (17)$$

$$P_j^* = \frac{d}{4c^2-2d^2}D_i + \frac{4c^2-d^2}{8c^3-4cd^2}D_j + \frac{d}{4c}C_i + \frac{d^2+8c^3-4cd^2}{8c^2-4d^2}C_j \quad (18)$$

$$Q_j^* = \frac{cd}{4c^2-2d^2}D_i + \frac{1}{2}D_j + \frac{d}{4}C_i + \frac{cd^2-8c^4+4c^2d^2}{8c^2-4d^2}C_j \quad (19)$$

$$P_s^* = \frac{1}{2a}\left[D_s + dP_s^{'} + \delta a C_s\right] \tag{20}$$

3.3 Mean Absolute Percentage Difference Rate of the Fuzzy Demand (MAPDR-FD)

To solve the two–echelon logistic model, we need to defuzzify the fuzzy demand first. We propose a defuzzification formula as follows.

Definition 4. Defuzzification Formula of Expected Value
Let $X = [a, b, c]$ be a continuous fuzzy value on U, which is the universal set. The defuzzification formula of expected value is defined as follows.

$$E[X] = E(o) + \left[1 - e^{-E(l)}\right],$$

where o is the central point and l is the radius which were defined in Definition 3.

It is straightforward to see that the function is a well-defined function because it satisfies the axioms for the order relations.

We also define a mean absolute percentage difference rate of fuzzy demand to assess the sensitivity of the results. The definition is as follows.

Definition 5: Mean Absolute Percentage Difference Rate of the Fuzzy Demand (MAPDR-FD)
The mean absolute percentage difference rate of the fuzzy demand (MAPDR-FD) is a measure of accuracy of a method for presenting the difference between fuzzy and non-fuzzy values. It usually expresses accuracy as a percentage, and is defined by the formula:

$$MAPDR - FD = \frac{100\%}{n}\sum_{k=1}^{n}\left|\frac{\pi_k^l - \pi_{F_k}^l}{\pi_k^l}\right|$$

where π_k^l is the non-fuzzy value and $\pi_{F_k}^l$ is the fuzzy value, where $l = i, j, s$ and $k = 1, 2, \dots, n$.

4 Numerical Examples

This section presents the numerical results to explain how using fuzzy case was better than using non-fuzzy case and illustrate how shippers, shippers' competitor and forwarders' decision were made by simulation in four shipping cases as the discussion in previous section. The assumed parameter values for these examples are listed in Table 1. First, fuzzy demand of both forwarders and a shipper is calculated. Then, three game behaviors were detected. Finally, MAPDR_FD was considered.

Table 1. Assumed parameters

Parameters	A	B	c	d	$P_s^{'}$	C_s	δ	β	θ	$\propto$
Values	0.5	0.4	0.5	0.4	30	2	1.2	1.5	1.4	0.01

4.1 Defuzzify Fuzzy Data

Let X_S be a fuzzy number of a shipper. The expected value of triangular fuzzy data X_S is given as follows.

$$E[X_s] = \big(E(a), E(b), E(c)\big) = (8{,}15{,}27).$$

By Definition 3, the expected value of central point is $\mathrm{E}(o) = \frac{8+15+27}{3} \approx 16.67$ and the expected value of radius is $\mathrm{E}(l) = \frac{27-8}{4} = 4.75$.

According to the Definition 4, the expected value of defuzzification formula is given as follow.

$$E[X_s] = E(o) + [1 - e^{-E(l)}] \approx 17.66$$

Similarly, we obtain the fuzzy demand of a shipper and forwarders for case 1 as shown in Table 2.

Table 2. Fuzzy demand of a shipper and forwarders

	Expected Value of Triangular Fuzzy Data	Defuzzification Data	
$\boldsymbol{X_s}$	(8,15,27)	17.66	$\boldsymbol{E[X_s] = D_s}$
$\boldsymbol{X_i}$	(9,20,29)	20.33	$\boldsymbol{E[X_i] = D_i}$
$\boldsymbol{X_j}$	(4,10,23)	13.32	$\boldsymbol{E[X_j] = D_j}$

We also shown the results of other cases and parameters we need in Table 3.

Table 3. Numerical data of four scenarios

	D_s	D_i	D_j	C_i	C_j
Case 1	17.66	20.33	13.32	1.75	2.00
Case 2	17.66	20.33	17.62	1.75	1.75
Case 3	17.66	17.66	13.32	2.00	2.00
Case 4	20.33	13.32	13.32	1.50	2.00

4.2 Comparison of the Optimal Solution for the Two-Echelon Model

By using the method of Cournot, Collusion and Stackelberg into the four cases, the demand for both the shippers and the forwarders was adjusted based on the requirement of the different cases. As a result of Table 4, Collusion behavior was the best solutions for the shippers gain the highest profit. This represents the shippers gained benefits from choosing cooperate with the downstream logistics providers. Stackelberg made the forwarders gain the highest profit, but the Collusion made the forwarders gain the lowest profit. The forwarder i got higher profit than the forwarder j when the forwarder i acted as a leader, and the forwarder j acted as a follower. Stackelberg decision was better than Cournot.

The quantity is proportional to the demand, if the demand of the forwarders is assumed to be smaller than the shippers. Both the forwarders act Stackelberg for getting bigger profits. The Collusion made the forwarders gain the lowest profit. In Stackelberg structure, the forwarder i got higher profit than the forwarder j when the forwarder i acted as a leader. The same principle happens when the forwarder j acted as a leader as well.

Table 4. Results of four scenarios

		P_i^*	P_j^*	P_s^*	Q_i	Q_j	Q_s	π_i^*	π_j^*	π_s^*
Case 1	Cournot	32.06	27.15	28.76	15.16	12.57	15.28	459.39	316.24	726.73
	Collusion	23.58	17.22	28.76	15.42	14.15	15.28	336.70	215.31	946.70
	Stackelberg	39.19	30.00	28.76	12.73	14.00	15.28	476.68	392.06	634.98
Case 2	Cournot	34.05	32.11	28.76	16.15	15.18	15.28	521.50	460.87	521.36
	Collusion	23.90	21.44	28.76	16.95	16.46	15.28	375.50	324.05	800.51
	Stackelberg	41.64	35.15	28.76	13.56	16.70	15.28	541.14	557.77	406.65
Case 3	Cournot	29.03	25.94	28.76	13.52	11.97	15.28	365.40	286.51	851.17
	Collusion	21.00	17.06	28.76	13.98	13.19	15.28	265.67	198.72	1035.28
	Stackelberg	35.39	28.48	28.76	11.35	13.24	15.28	379.15	350.66	775.02
Case 4	Cournot	23.58	23.76	31.43	11.04	10.88	16.61	243.69	236.65	1360.66
	Collusion	16.40	16.40	31.43	11.69	11.69	16.61	174.06	168.22	1496.12
	Stackelberg	28.77	25.83	31.43	9.27	11.92	16.61	252.87	284.01	1304.67

4.3 Comparison Results of Fuzzy Case and Non-Fuzzy Case

The mean absolute percentage difference rate of the fuzzy demand was calculated to illustrate the difference between crisp number and fuzzy results. As a result that is shown in Table 5, the results by using fuzzy variables gained higher profits that the results by using non-fuzzy variables. After calculating the mean absolute percentage difference rate of fuzzy demand, it demonstrated percentage that results of fuzzy differ from results of non-fuzzy. MAPRD_FD clarifies how far distance between crisp number and fuzzy data. In some case, the fuzzy profits were lower than non-fuzzy case owing to fuzzy demand was less than crisp demand. It happened only profits of the shipper because after computing fuzzy demand of shipper, the answer was lower than real data.

Table 5. Results of MAPDR_FD

		Case 1	Case 2	Case 3	Case 4
		MAPRD_FD	MAPRD_FD	MAPRD_FD	MAPRD_FD
Profit i	Cournot	14.40%	10.95%	47.14%	78.25%
	Collusion	13.04%	9.89%	46.04%	77.99%
	Stackelberg	3.71%	1.87%	31.13%	78.25%
Profit j	Cournot	42.93%	25.61%	63.62%	80.03%
	Collusion	48.58%	27.48%	65.92%	80.05%
	Stackelberg	70.77%	46.50%	92.99%	79.88%
Profit s	Cournot	17.83%	29.70%	3.68%	11.92%
	Collusion	19.32%	25.15%	10.14%	7.63%
	Stackelberg	15.06%	31.91%	0.03%	13.80%

5 Discussion and Conclusion

5.1 Discussion

Comparing the data in Tables 4 and 5, we could see that it could be suggested that the optimal solution for shipper should act according to Collusion behavior with downstream LSPs, in this case was forwarders. We also had large difference between fuzzy and non-fuzzy data in this case. It means that although we had large profit but we should also consider the risk in this problem.

However, if the shipper was willing to act as Stackelberg, forwarders could contribute some profit. For forwarders, Stackelberg behavior was a favorable choice. The results also concluded that the fuzzy profits yield higher than the profits which used crisp number.

The MAPDR_FD reminded us how much risk we should consider in this problem. It also could clarify the results between fuzzy demand and non-fuzzy demand that show us how far between both of them. Therefore, the results could be described that the fuzzy results were differentiate from the crisp number's results in this problem.

5.2 Conclusion and Future Works

In this research, we considered fuzzy demand instead of crisp demand and illustrated three competitive behaviors, Cournot, Collusion, and Stackelberg in the two-echelon logistic system. We analyzed the competitive and cooperated action describing under the game approach and translated the theory to mathematical results. Moreover, the profits of fuzzy data were bigger than the profits of non-fuzzy data. Finally, the mean absolute difference percentage of fuzzy demand was used to clarify how difference between the results of non-fuzzy case and fuzzy case.

In summary, the optimal solution for shipper should act according to Collusion behavior with downstream LSPs, in this case is forwarders. However, if the shipper was willing to act as Stackelberg, forwarders could contribute some profit. For forwarders, Stackelberg behavior was a favorable choice. The results also concluded that we got higher profits by using fuzzy numbers.

As the limitation, the mathematical models illustrated the competed situation with a small number of competitors. Therefore, several factors such as location, operational efficiency, shipping distance, and reputation are still neglected. As further studies, we would like to combine fuzzy random variables and the probability in the model and consider a multi-stage year based on real option in order to forecast the profits of the shipper and forwarders in multi-year.

References

1. Yanga, S.-L., Zhou, Y.-W.: Two-echelon supply chain models: Considering duopolistic retailers' different competitive behaviors. Int. J. Production Economics, 104–116 (2006)
2. Von Neumann, J., Morgenstern, O.: Theory of Games and Economic Behavior, 2nd edn. (1st edn. 1944). Princeton University Press, Princeton (1947)

3. Lau, A., Lau, H.-S.: Effects of a demand-curve's shape on the optimal solutions of a multi-echelon inventory pricing model. European Journal of Operational Research 147, 530–548 (2002)
4. Lau, A., Lau, H.-S.: Some two-echelon supply-chain games: improving from deterministic-symmetric-information to stochastic asymmetric-formation models. European Journal of Operational Research 161(1), 203–223 (2005)
5. Thisana, W., Rosarin, D., Watada, J.: Optimal Decision Methods in Two-echelon Logistic Models. Journal of Management Decision (2012)
6. Liu, B., Liu, Y.: Expected Value of Fuzzy Variable and Fuzzy Expected Value Models. IEEE Transactions on Fuzzy Systems 10(4), 445–450 (2002)
7. Liu, B.: A survey of credibility theory. Fuzzy Optimization and Decision Making 5, 387–408 (2006)
8. Puri, M.L., Ralescu, D.A.: Fuzzy Random Variables. J. Math. Anal. Appl. 114(2), 409–422 (1986)
9. Nahmias, S.: Fuzzy variables. Fuzzy Sets and Systems 1, 97–110 (1978)
10. Nureize, A., Watada, J.: Fuzzy Random Regression Based Multi-attribute Evaluation and Its Application to Oil Palm Fruit Grading. Annals of Operation Research, 1–17 (2011)
11. Nureize, A., Watada, J.: Building Fuzzy Goal Programming with Fuzzy Random Linear Programming for Multi-level Multi-Objective Problem. International Journal of New Computer Architectures and their Applications 1(4), 911–925 (2012)
12. Chenxi, Z., Ruiqing, Z., Wansheng, T.: Two-echelon supply chain games in a fuzzy environment. Computers and Industrial Engineering 55, 390–405 (2008)
13. Lin, P.-C., Watada, J., Wu, B.: Risk Assessment of a Portfolio Selection Model Based on a Fuzzy Statistical Test. IEICE Transactions on Information and Systems E96-D(3), 579–588 (2013)
14. Lin, P.-C., Wang, S., Watada, J.: Decision Making of Facility Locations Based on Fuzzy Probability Distribution Function. In: The IEEE International Conference on Industrial Engineering and Engineering Management (IEEM 2010), Macao, China, pp. 1911–1915 (2010)
15. Lin, P.-C., Watada, J., Wu, B.: A Parametric Assessment Approach to Solving Facility Location Problems with Fuzzy Demands. IEEJ. Transactions on Electronics, Information and Systems 9(5) (September 2014)
16. Lin, P.-C., Watada, J., Wu, B.: Identifying the Distribution Difference between Two Populations of Fuzzy Data Based on a Nonparametric Statistical Method. IEEJ. Transactions on Electronics, Information and Systems 8(6), 591–598 (2013)
17. Lin, P.-C., Wu, B., Watada, J.: Kolmogorov-Smirnov Two Sample Test with Continuous Fuzzy Data. Integrated Uncertainty Management and Applications 68, 175–186 (2010)

A Hybrid Approach to Modelling the Climate Change Effects on Malaysia's Oil Palm Yield at the Regional Scale

Subana Shanmuganathan[1,2], Ajit Narayanan[1,3], Maryati Mohamed[4,6],
Rosziati Ibrahim[5,6], and Haron Khalid[7]

[1] Auckland University of Technology (AUT), New Zealand
[2] Geoinformatics Research Centre (GRC)
[3] School of Computer and Mathematical Sciences (SCMS)
[4] Department of Technology and Human Development, Faculty of Science
[5] Research, Innovation, Commercialisation & Consultancy Office (RICCO)
[6] Universiti Tun Hussein Onn Malaysia (UTHM), Johor, Malaysia
[7] MPOB Research Station Kluang, Malaysia
{subana.shanmuganathan,ajit.narayanan}@aut.ac.nz,
{maryati,rosziati}@uthm.edu.my, khalid@mpob.gov.my

Abstract. Understanding the climate change effects on local crops is vital for adapting new cultivation practices and assuring world food security. Given the volume of palm oil produced in Malaysia, climate change effects on oil palm phenology and fruit production have greater implications at both local and international scenes. In this context, the paper looks at analysing the recent climate change effects on oil palm yield within a five year period (2007-2011) at the regional scale. The hybrid approach of data mining techniques (association rules) and statistical analyses (regression) used in this research reveal new insights on the effects of climate change on oil palm yield within this small data set insufficient for conventional analyses on their own.

Keywords: Regression test, Data mining (association rules), WEKA, JRip.

1 Introduction

Climate change has the potential to impact on almost all forms of agricultural systems and their productivity at varying degrees e.g., from small benefits to extensive drastic effects depending on the current climate regime and the crop [1]. Understanding this variability in climate change and its effects on local crops is vital for adapting new cultivation practices to overcome the harsh effects on crops, especially to assure word food security now and in the future. Malaysia's oil palm exports contribute to 45% of the world's edible oil needs [2]. Hence, it is essential that we understand the climate change and its potential effects under different scenarios for palm oil production. The paper looks at the correlations between recent climate change and oil palm yield in Malaysia's administrative regions over a period of five years (2007-2011) to

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_32, © Springer International Publishing Switzerland 2014

determine the trends and possible climate change effects. The work presented here is an extension to earlier work that looked at possible climate change effects in West (Peninsular) and East (Boneo) Malaysia, outlined in section 2.3 of this paper.

2 Previous Research

Previous research results provide clues on the phonological stages that are susceptible to the extreme climate events and are outlined in this section.

2.1 Effects of El Niño and La Niña in Tumaco, Colombia

In a pioneering study [3], the effects El Niño and La Niña events of in the local climate and on oil palm tree (Elaeis guineensis) production in the Tumaco situated in the Pacific Coast of Colombia were reported. In that study, it was established that El Niño and La Niña events had lagged (previous period) and conflicting impacts on oil palm yields. El Niño experienced in 1997/98 was found to be favourable showing the maximum correlation with production 2.6 years after the event. Meanwhile, La Niña of 1999/2000 had caused severe droughts, with the highest negative impact (reduced yield) in 2002.

2.2 Effects of El Nino Events in Sabah, Malaysia

In [4] using statistical analyses, the authors established that heavy rains and high temperatures as favourable to palm oil production in the western coast of Sabah with a lag period of 3 and 4 months respectively. However, the extremes of both scenarios, i.e. flooding and severe drought, were found to be not necessarily favourable to oil palm production. The extremes of La Niña events, e.g. higher precipitation/floods, decreased the production and quality of crude palm oil (CPO), the events being attributed to affecting the fruit ripening stage (fig. 6) reflected in the yield at later months. The Sabah study [4] looked at the climate variability in rainfall, temperature and their correlations with fresh fruit bunch (FFB) production and fish landings in the West coast of Sabah, Malaysia, using data from 2000 to 2010.

2.3 Climate Change Effects on Oil Palm in Malaysia

In view of the above two studies in [5] an attempt was made to model the climate change variability and its effects on oil palm yield in the East (Peninsular) and West (Boneo/ Sabah and Sarawak) Malaysia located on either side of the South China Sea. The results of that initial research using monthly climate anomalies in temperature and oil palm yield from the two main regions of Malaysia revealed new insights on the correlations between the climate variations, oil palm tree phenology and yield. The climate variability in the two main Malaysian regions and the lagged effects of temperature on the oil palm yield conformed to the phenological growth stages of the oil palm tree, established as vulnerable to climate change in earlier work in the

Columbian and Malaysian contexts (sections 2.1 and 2.2 respectively). However, it was found that further research was required to establish the precise effects of temperature and the phenological stages vulnerable to temperature, using long term yield and weather data especially at the regional scale.

The aim of this paper is to extend our growing knowledge of lag periods and their effects on palm oil growth and production. In particular, there are still questions on how changes in lag period suggested in the earlier studies, are related to production yield. Such knowledge is critical for economic and production planning purposes so that, after a severe climate event, appropriate plans and actions can be put in place to limit the effects of the severe event at suitable time points subsequently.

In the next section the methodology and data used in this research are discussed.

3 The Methodology

For this research, monthly averages of temperature and oil palm yield (Fresh Fruit Bunch/ FFB) data at the regional scale were used for analysing the correlations between the climate and yield at this scale. The data set is analysed using a hybrid approach consisting of Top-Down Induction of Decision Trees (e.g. JRip function in WEKA software) and statistical regressions. Two sets of regressions were run, firstly, one test with all regional yield and lag variables together and then 10 more regressions were run with each of the individual regional data separately. The WEKA rules (generated for all regions) and regression results of both all regions together and the 10 individual regions alone were compared to investigate the trends in the climate change effects on oil palm yield across Malaysia in general and those specific to the individual regions.

3.1 Oil Palm Yield Data

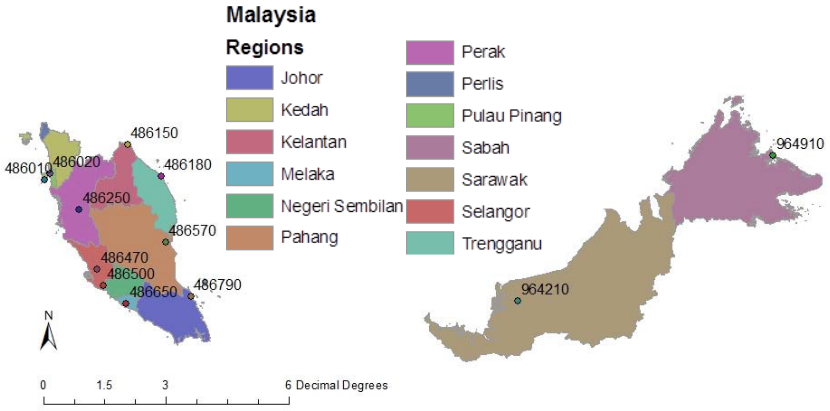


Fig. 1. Map of Malaysia's administrative regions and weather stations (table 1) [7]

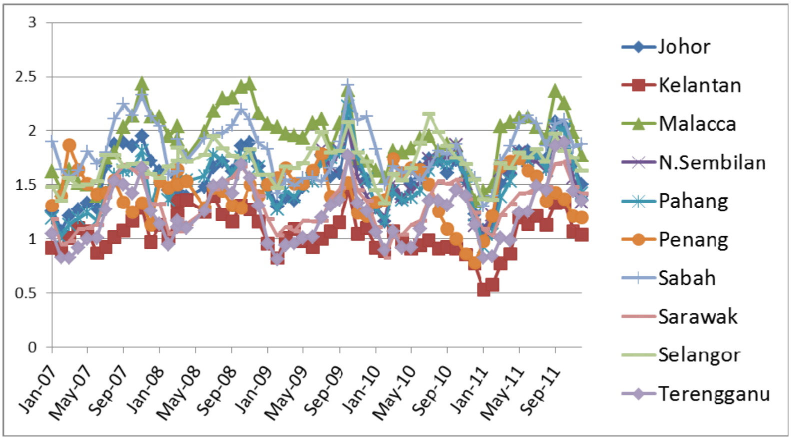


Fig. 2. Graphs showing oil palm yield (fresh fruit bunch/FFB in T/H/M) in the ten Malaysian regions studied in the research

Table 1. Yield regions and weather stations (global daily summary temperature) data was obtained. Source: NCDC Climate Services Branch (CSB) [7].

Sta no.	Station name	Yield region
486010	PENANG/BAYAN LEPAS	Penang
486020	BUTTERWORTH	Kedah
486150	KOTA BHARU	Kelantan
486180	KUALA TRENGGANU	Terengganu
486250	IPOH	Perak
486470	KL SUBANG	Selangor
486500	KUALA LUMPUR INTL N.	Sembilan
486570	KUANTAN	Pahang
486650	MALACCA	Malacca
486790	JOHORE BHARU/SENAI	Johor
964210	SIBU	Sabah
964910	SANDAKAN	Sarawak

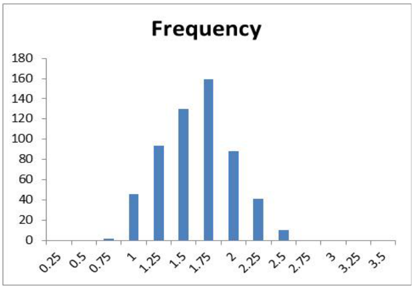


Fig. 3. Oil palm yield frequency distribution within 2007-2011 monthly regional yield data. WEKA yield classes (FFB in T/H/M) Low <1.25, Medium >1.25 <2, and High>2.

Monthly oil palm yield (FFB) data for the regions (fig. 1) for a period of five years (2007-2011) obtained from [6], presented as graphs (fig. 2) was analysed against 36 lag variables (36 monthly average temperatures prior to harvest).

3.2 Yield Classes

Based on the data distribution and frequency histogram (fig. 3), yield **(FFB in T/H/M)** was classified into three classes for analysis using supervised learning in WEKA: Low <1.25, Medium /Med ≥1.25 <2, and High≥2.

3.3 Climate Data

Global daily temperature summaries extracted from [7] for stations (fig 1 and table 1) in the respective regions were converted into a table of monthly average temperatures (^{o}C) for the corresponding 10 regions of Malaysia studied in this research.

In this research, 36 monthly temperature averages calculated for the 10 regional stations (fig. 1 and Table 1), and the monthly oil palm yield for these individual regions (fig. 2) were analysed to determine the lag effects of the climate on oil palm yield across Malaysia at the regional scale and the results are presented in next section

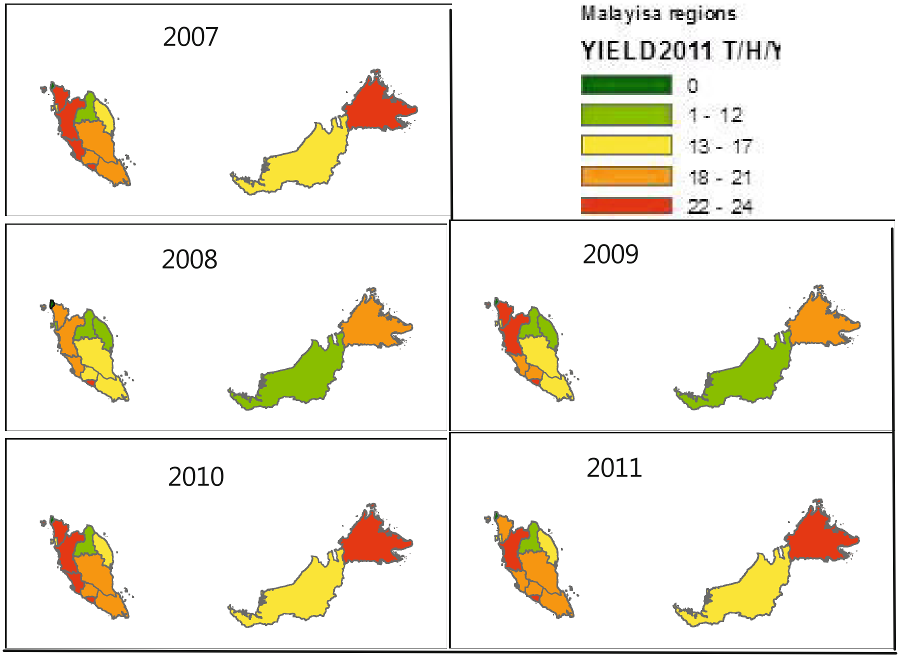


Fig. 4. Annual Oil Palm yield (T/H/Y) variability during 2007-11

4 The Results

The variability in regional monthly and annual yield appears to have some common patterns across Malaysia and some other specific characteristics unique to the individual regions (figs. 2 and 4). For example, the yield is high mostly in the north western regions of Peninsular Malaysia and in Sabah and this can be observed throughout the study period 2007-2011 (fig 4).

```
JRIP rules:
1. (state = Malacca) and (lag7 <= 27.55) and (lag10 <= 27.35)
        => yield class=high (14.0/1.0)
2. (state = Sabah) and (lag28 >= 26.86) => yield class=high (13.0/4.0)
3. (state = Kelantan) => yield class=low (60.0/10.0)
4. (lag33 >= 27.53) and (state = Sarawak) => yield class=low (28.0/6.0)
5. (state = Terengganu) and (lag2 <= 27.28) => yield class=low (23.0/2.0)
6. (lag8 >= 27.83) and (lag23 <= 26.62) => yield class=low (16.0/5.0)
7. (lag21 >= 27.29) and (state = Terengganu) and (lag16 <= 26.77)
        => yield class=low (7.0/0.0)
8. => yield class=med (409.0/56.0)
Number of Rules: 8
Time taken to build model: 0.32 seconds
=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances        450      78.9474 %
Incorrectly Classified Instances      120      21.0526 %
Kappa statistic                       0.5441
Mean absolute error                   0.1918
Root mean squared error               0.3408
Relative absolute error               59.0095 %
Root relative squared error           84.616 %
Total Number of Instances 570
=== Detailed Accuracy By Class ===
TP  Rate  FP Rate  Precision    Recall  F-Measure   ROC Area   Class
    0.882 0.397    0.818        0.882   0.848       0.755      med
    0.674 0.056    0.795        0.674   0.729       0.836      low
    0.412 0.04     0.5          0.412   0.452       0.747      high
WA. 0.789 0.282    0.784        0.789   0.784       0.774
=== Confusion Matrix ===
a      b      c <-- classified as
336    24     21 | a = med
45     93      0 | b = low
30     0      21 | c = high
```

Fig. 5. JRip (WEKA rules) generated using yield classes (low, medium and high) as dependent and lag variables (36 months prior to harvest) as independent factors

4.1 WEKA Results

The WEKA JRip rule function run with 36 monthly average temperatures prior to harvest (lag variables relating to berry development cycle) and the monthly yield of all 10 regions produced association rules that classified the yield classes at 78.9% accuracy (fig 5)**.** The 7 rules (fig. 5) produced by the WEKA function showed the critical monthly temperatures that had had lagged effects on the yield in five specific regions (Malacca, Sabah, Kelantan, Sarawak and Terengganu), as well as across Malaysia.

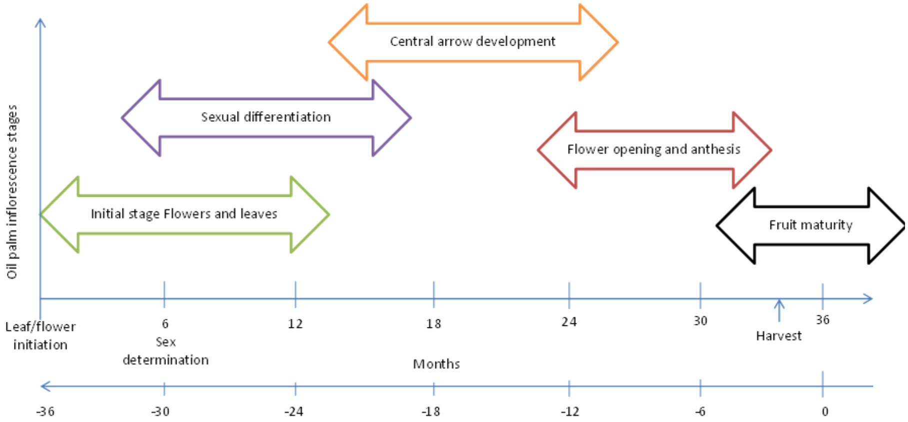


Fig. 6. Oil palm life cycle (frond emergence month 1/36 to -36/0 to fruit harvest) with stages and approximate timing relating to bunch development

The first two JRip rules (fig. 5) showed the lagged effects of temperature that had led to “high” (>2 T/H/M) yield in Malacca (14 instances/ 1 exception) and Sabah (13 instances/ 4 exceptions). For the Malacca region, temperatures in month -7 (≤ 27.55°C) and -10 (≤ 27.35 °C) had led to “high” yield. Similarly, for Sabah temperature in month -28 (≥ 26.86 °C) had led to “high” yield. In oil palm phenology months -7 to -10 are the time when flower opening and anthesis occur, and month -28 relates to the initial frond emergence and sex determination in the berry development cycle (fig. 6).

Rule 3 (fig. 5 with 60 instances/10 exceptions) implies that the yield in Kelatan is “low” when compared with that of the other regions. Based on rule 4, in Sarawak, lag variable/ month -33 ≥ 27.53°C temperature had led to “low” yield, and this relates to the frond emergence phase (fig. 6).

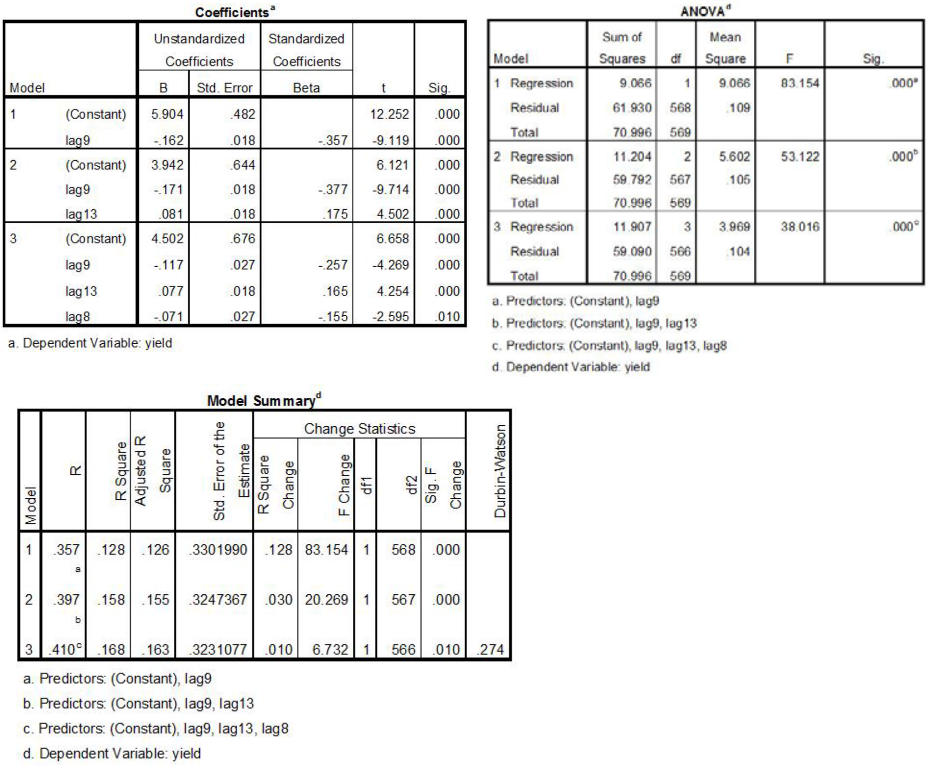
Based on rule 5, in Terengganu, lag variable, month -2 ≤27.28°C had led to “low” yield, and this relates to the fruit maturity period (fig. 6).

Rule 6 for all regions relates to lag variable month -8 ≥ 27.83°C and month -23 ≤ 26.62°C that had led to “low” yield with 16 instances and 5 exceptions. This implies that higher temperatures during anthesis/ early fruit development and lower temperatures during frond /leaf emergence had led to “low” yield across the country.

Based on rule 7, in Terengganu lag variable month -21 ≥ 27.29°C and month -16≤ 26.77°C had led to “low” yield, met in 7 instances. Hence, in Terengganu also, high/ low temperatures during flower opening and sex determination stage had affected on the region’s yield (fig. 6).

4.2 Regression Test Results

From the first regression test run on all 10 regional data together, month −9, −13 and −8 temperatures were shown as the most “critical lag variables” or predictors for the monthly yield (fig. 7). Earlier studies from literature [2][4] and authors of this paper [5] established the phenological stages relating to these months as: flower opening and anthesis (month -8/-9), and sexual determination month -13 (fig. 6).



Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients		
		B	Std. Error	Beta	t	Sig.
1	(Constant)	5.904	.482		12.252	.000
	lag9	-.162	.018	-.357	-9.119	.000
2	(Constant)	3.942	.644		6.121	.000
	lag9	-.171	.018	-.377	-9.714	.000
	lag13	.081	.018	.175	4.502	.000
3	(Constant)	4.502	.676		6.658	.000
	lag9	-.117	.027	-.257	-4.269	.000
	lag13	.077	.018	.165	4.254	.000
	lag8	-.071	.027	-.155	-2.595	.010

a. Dependent Variable: yield

ANOVA[d]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	9.066	1	9.066	83.154	.000[a]
	Residual	61.930	568	.109		
	Total	70.996	569			
2	Regression	11.204	2	5.602	53.122	.000[b]
	Residual	59.792	567	.105		
	Total	70.996	569			
3	Regression	11.907	3	3.969	38.016	.000[c]
	Residual	59.090	566	.104		
	Total	70.996	569			

a. Predictors: (Constant), lag9

b. Predictors: (Constant), lag9, lag13

c. Predictors: (Constant), lag9, lag13, lag8

d. Dependent Variable: yield

Model Summary[d]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	.357[a]	.128	.126	.3301990	.128	83.154	1	568	.000	
2	.397[b]	.158	.155	.3247367	.030	20.269	1	567	.000	
3	.410[c]	.168	.163	.3231077	.010	6.732	1	566	.010	.274

a. Predictors: (Constant), lag9

b. Predictors: (Constant), lag9, lag13

c. Predictors: (Constant), lag9, lag13, lag8

d. Dependent Variable: yield

Fig. 7. The regression results of all 10 regional yields and their lag variables run together. The predictors (lag variables) average temperatures of month -9, -13 and -8 correspond to flower opening and anthesis of oil palm phenology (fig. 6).

The 10 different regressions run on 36 lag variables against the respective regional yield individually produced the respective predictor/s (lag variables/ monthly average temperatures) for the yield at the regional scale (table 2). In the regions studied, month -7 or -8 or -9 or a combination of these monthly temperatures were shown to be having lagged effects on the yield and this -7 to -9 month (or prior to harvest) period corresponds to flower opening and anthesis stage (fig. 6).

For Kelantan, N. Sembilan, Pahang, Malacca, Johor and Sabah the critical temperature is in month -9. Hence, this is one of the predictors for yield. These regions are found in the southern Peninsula Malaysia and northern Boneo Malaysia except for Kelantan on the north east tip. Similarly, for Selangor and Penang it is month -8. Sarawak -10 and for Terengganu -7 are the months found to be among the predictors. Malacca has -9 as the sole predictor for the region's yield, and again this is the period flower opening and anthesis occur. It seems that in all the regions temperature affects the flower opening and anthesis, with their lagged effects reflected on each of the region's yield in 7-10 months later.

Table 2. The regions, months and R2 that showed lagged effects on yield. (-27 indicates the model without the lag variable average temperature of month 27.

Station No		Region	lag variable/ month- R square
486010	Penang	**23, 27, 6, 8, 31**	0.43 0.50 0.59 0.59 0.63 0.71
486020	Kedah	---	
486150	Kelantan	**9, 4**	0.28 0.39
486180	Terengganu	**33, 27, 21, 7, 15, 9**	0.45 0.63 0.69 0.73 0.75 0.78 0.78 (-27
486250	Perak	---	
486470	Selangor	**8, 29, 20, 33, 17 15**	0.30 0.49 0.59 0.63 0.65 0.68
486500	N. Sembilan	**9, 31, 35, 1, 3**	0.45 0.60 0.67 0.72 0.79
486570	Pahang	**9, 13, 29, 33, 25**	0.51 0.67 0.72 0.75 0.79
486650	Malacca	**9,**	0.27
486790	Johore	**9, 2, 29, 13**	0.33 0.47 0.66 0.73
964210	Sabah	**9, 14, 17, 13, 31**	0.38 0.46 0.54 0.60 0.62
964910	Sarawak	**33, 27, 13, 29, 7, 10, 14**	0.50 0.66 0.72 0.77 0.80 0.82 **0.84 0.84** (-27

The average temperatures of month 29-35 prior to harvest as well showed lagged effects on the yield in 8 of the 10 regions (table 2), which is the frond/leaf emergence stage, also confirmed by JRip rule 6 (fig 6). The other stage shown as critical in the JRip rule 6 is the month -8, which is the flower anthesis phase in the oil palm berry bunch development cycle, also confirmed by all regressions (all regional data together and individual regions separately). Meanwhile, in Sarawak and Terengganu regions, frond emergence stage (month -33) seems be the primary predictor for yield.

The regressions run on individual regional data separately gave higher adjusted R square values for the models, 0.27 to maximum of 0.84 as opposed to 0.128-0.168 in the all regions together. This shows that the influence exerted by the lag variables on the yield varies for the different regions but for the same phenological phase (flower opening and anthesis at 7-13 months prior to harvest) hence, the effects cannot be generalised across regions. Further analysis incorporating local data (e.g. rainfall, terrain) could shed further insights on the precise climate change effects on yield.

5 Conclusions

The paper looked at the possible climate change effects on Malaysia's oil palm yield using 36 monthly average temperatures as lag variables along with yield data at the regional scale. Even with this small data set from 2007 to 2011, this initial investigation conducted at this scale has provided some useful insights on the variability in the lag effects of temperature on the monthly oil palm yield. For instance, based on JRip rule 6 lag variable, average monthly temperature -8 (8 months prior to harvest) ≥ 27.83°C has led to "low" yield (<1.25 T/H/M) across all regions of Malaysia.

Rules 2 and 6 indicate that subtle increases e.g., 27-28°C in monthly average temperatures at the initial stages of fruit development, have led to "high" yields in Sabah.

From the regression run on all regional yields together, lag variables average temperatures of month -9, -13 and -8 (flower opening and anthesis) were found to be the predictors. This has also been reflected in the individual regional regressions but the predictability of the lag variables were much higher. This implies that in all the regions the effects of temperature in the early stages of the fruit development are lagged and reflected in the yield in 8-13 months later.

The adjusted R square value for all 10 regions together regression model (month -9, -13 and -8) was low (0.128-0.168 as opposed to the 0.27-0.84 of the individual regional regression models) and this indicates that the effects of climate change on oil palm yield for the whole nation could not be generalised.

In conclusion, this research indicates that attempting to predict the effects of climate change on palm oil yield in particular, and perhaps other crops more generally, will need to be regionalised if accuracy of prediction is required. The exact size of region will depend on local environmental conditions. Further research at finer scales, such as regional or blocks that are topographically discrete, is warranted to establish the climate change effects at the "*meso*" and micro scales (among and within oil palm plantations).

References

1. Iglesias, A., Rosenzweig, C.: Effects of Climate Change on Global Food Production, Special Report on Emission Scenarios (2009), `http://sedac.ciesin.columbia.edu/mva/cropclimate/`
2. USDA-FAS Office of Global Analysis, Indonesia: Palm Oil Production Prospects Continue to Grow, `http://www.pecad.fas.usda.gov/highlights/2007/12/Indonesia_palmoil/`
3. Cadena, M.C., Devis-Morales, A., Pabón, J.D., Málikov, I., Reyna-Moreno, J.A., Ortiz, J.R.: Relationship between the 1997/98 El Niño and 1999/2001 La Niña events and oil palm tree production in Tumaco, Southwestern Colombia. Adv. Geosci. 6, 195–199 (2006)
4. Wen, P.P., Sidik, M.J.: Impactsof Rainfall, Temperature and Recent El Niños on Fisheries and Agricultural Products in the West Coast of Sabah 2000-10. Borneo Science 28, 73–85 (2011)

5. Shanmuganathan, S., Narayanan, A.: Modelling the climate change effects on Malaysia's oil palm yield. In: Proceedings of 2012 IEEE Symposium on IS3e 2012, Kuala Lumpur, Malaysia, October 21-24 (2012)
6. Malaysia Oil Palm Board, "Statistics-Yield," Economics and Industry Development Division, Kelana Jaya Selangor
7. NOAA (2012), `http://www.ncdc.noaa.gov/cdo-web/`

A New Algorithm for Incremental Web Page Clustering Based on k-Means and Ant Colony Optimization

Yasmina Boughachiche[1] and Nadjet Kamel[1,2]

[1] LRIA, Department of Computer Science
USTHB, Algiers, Algeria
yboughachiche@usthb.dz
[2] University of Setif 1, Setif, Algeria
nkamel@usthb.dz

Abstract. Internet serves as source of information. Clustering web pages is needed to identify topics in a page. But dynamism is one of the web clustering challenges, because the web pages change very frequently and new pages are always added and removed. Processing a new page should not require to repeat the whole clustering. For these reasons, incremental algorithms are an appropriate alternative for web page clustering

In this paper we propose a new hybrid technique we call Incremental K Ant Colony Clustering (IKACC). It is based on the Ant Colony Optimization and the k-means algorithms. We adapt this approach to classify the new pages in the online manner, and we compare it to incremental k-means algorithm. The results show that this approach is more efficient and produces better results.

Keywords: Incremental Clustering, Ant colony optimization, k-means, web content mining

1 Introduction

Clustering algorithms aim at grouping data into groups such as similar data is assigned to the same group. These algorithms are used in many applications such as web pages clustering to identify the topics of a page. Most of the existing clustering methods process statically on the whole collection of documents (i.e., all data are collected before clustering) in order to produce fixed number of clusters. This can not be efficient with the web because the Web data is Huge in amount, distributed, heterogeneous, unstructured or semi structured with different forms like html or xml, and dynamic. Adding or removing a new page should not require to repeat the whole clustering process. For these reasons, incremental clustering algorithms are an appropriate alternative for web page clustering. This kind of incremental clustering techniques is also called online clustering.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_33, © Springer International Publishing Switzerland 2014

In this paper we propose a new hybrid clustering algorithm. This algorithm is based on the hybridization of the K-means clustering algorithm [1] and the ant colony algorithm [2]. First, the Ant clustering algorithm is used to group the data into clusters and place them on a 2D grid. Then, the k-means algorithm is applied. This reduces the number of total clusters by merging related results. To overcome the problem of the dynamic nature of web, we propose a third step. We modify the ant colony and the k-means clustering algorithm to cluster the new added pages. Furthermore, as a measure of the performance of the algorithm, we offer an extended set of experimental results using differing test-sets and compare the algorithm against the incremental k-means clustering.

The rest of this paper is organized as follows: Section 2 gives an overview on incremental clustering algorithms. Section 3 lays out our new hybrid clustering algorithm ”the Incremental K-Ant Colony Clustering algorithm”. Experimental results are illustrated in section 4. Section 5 gives a discussion on the IKACC method. Finally, in section 6 we present our conclusion and future works.

2 Related Works

Incremental clustering is a technique developed to avoid the problem of re-clustering data [3]. Many works have been proposed to deal with dynamic web data, proposing incremental clustering algorithms. Gavin and Yue [4] focused on good feature extraction and document representation and a good clustering approach.

Ant based clustering found success in solving incremental clustering problems. The algorithms based on ACO have the ability to create clusters of objects without any initial partitioning, and without knowing a priori how many clusters are necessary. Also, ant based algorithms are suitable for dynamic data, and can adapt the placement of existing clusters, forming new clusters or deleting existing one. Some existing approaches as in [5] [6] [7] use the robustness and adaptivity of ant-based clustering algorithms, and may be able to deal with dynamic data.

Hybrid approaches combined with standard clustering algorithms are also a promising way for online clustering. In [8] a new dynamic data clustering algorithm based on K-means and particle swarm optimization(PSO), called KCPSO is proposed. In [9] integrated PSO with K-means are used to cluster data. Some other hybrid heuristic methods like genetic, simulated annealing and tabu-search with k-means were ever used with clustering algorithm to solve local optimal problem [10][11][12]. Moreover, the Ant colony Optimization was combined with k-means clustering algorithm to solve the incremental clustering problem, such as in[2][9][13].

3 Our Proposition

In this section, we propose our approach we call Incremental K Ant Colony Clustering (IKACC). It combines the Ant-based clustering, directed by entropy, and K-means clustering. It is composed of the following three main modules:

3.1 Data Pre-processing Module

This module extracts the word vectors of each web page, and produces the index for the classification step. The traditional method of terms weighting, widely used, is not suitable for the web page clustering. Furthermore, the diagram of term weighting can be used. It is based on either the frequency of occurrence of terms TF (Term Frequency), or on the frequency of occurrence of terms combined with the inverse of the frequency of the document TF-IDF (Term Frequency-Inverse Document Frequency). However, from the experiments, it was found that the web pages usually contain a small amount of words. Each word appears once or twice at most. In this case, the frequency of occurrence of a word is not possible to assess its importance in the document. So we use a new feature extraction for clustering web documents that do not depend on the term frequency method. This method makes a compromise in order to balance between the cover and the number of features used for the representation of documents[14].

We represent D_i document as follows: $D_i = (W_i, ID_i)$, where ID_i is the identifier of the document D_i and W_i is the feature vector of the document: $W_i = (w_{i1}, w_{i2}, ..., w_{iN})$. N represents the number of extracted features, and w_{ij} is the weight of the j^{th} feature, where $j \in 1, 2, ..., N$.

The selected weight is binary since $w_{ij} = 1$ if D_i contains the j^{th} feature, otherwise, $w_{ij} = 0$. Since a web page does not contain a lot of words, the frequency of words may not indicate the importance of this word. So this pattern of binary weight is more suitable to our problem.

For reasons of simplicity, the feature vector is constructed from only the text that is visible on the screen, like normal text, captions to images and text links.

3.2 The Clustering Module

This module produces classes from a set of available pages, using the hybridization of Ant Colony clustering and the k-means clustering algorithm. This is done through two steps:

First Step: In the initial state (t=t_0), select N_0 pages from the available, and distribute the N_0 objects uniformly randomly on the $Z \times Z$ 2D grid. Initialize all agents to be unladen (not carrying any object). Denote $s \times s$ as the region in which an agent lies. The entropy of the $s \times s$ area including a set of objects is defined by equation (1), where $p(x)$ is defined by equation (2), obj-num is the total number of pages in $s \times s$, x-num is the number of objects whose attribute X_i has value x.

$$E(s) = -\sum_{(i=1)}^{n} \sum_{x \in X_i} p(x) \log p(x) \tag{1}$$

$$p(x) = \frac{x - num}{obj - num} \tag{2}$$

According to [5], a document is considered as a probability distribution of terms. The similarity of two documents is measured as the distance between the two corresponding probability distributions. The relative entropy is a widely applied measure for evaluating the differences between two probability distributions.

The following algorithm is for preparing the next step of clustering. It Takes entropy as the criterion for an agent to pick up or drop items:

Algorithm 1. ant-cluster

```
Initialize parameters Z, s, $t_{max}$, $N_a$
for all object$o_i \in N_0$ do
    Place $o_i$ randomly on the plane of $Z \times Z$
end for
for n=1 to $N_a$ do                                    ▷ $N_a$ is the number of agents
    Place agent at randomly selected site in $Z \times Z$
end for
for t=1 to $t_{max}$ do               ▷ $t_{max}$ is the maximal times that an agent moves
    for n=1 to $N_a$ do
        if (agent unladen) and (site occupied by object $o_i$) then
            Compute entropy E1, E2
            if $E1 \leq E2$ then
                pick up $o_i$                                        ▷ picking up rule
            end if
        else if (agent carrying object $o_i$) and (site empty) then
            Compute entropy E1, E2
            if $E1 \geq E2$ then
                drop $o_i$                                           ▷ dropping rule
            end if
        end if
        Move to randomly selected neighbor site not occupied by other agent
    end for
end for
for all site (x,y) in $Z \times Z$ do
    Compute entropy of the surrounding area $s \times s$
    Compute pheromone $\tau(x, y)$
end for
```

E1 and E2 are the entropies before and after performing the relevant action In this algorithm, when all agents stop moving, the initial clusters have been formed. For the next incremental clustering, to each pixel (x,y) in the $Z \times Z$ plane, compute the entropy of the surrounding $s \times s$ area according to equation (1), and if $s \times s$ area is empty set entropy of the area with a maximum. Then compute the pheromone concentration $\tau(x, y)$ of the surrounding $s \times s$ area according to equation (3).

$$\tau(x, y) = (obj - num)/(1 + s \times s) \tag{3}$$

Where $(obj - num)$ is the number of objects in the surrounding $s \times s$ area. The stop rule is mainly based on a threshold (t_{max}) for the number of iterations.

The second step: after the pages are positioned on a two dimensions plane by ant colony algorithm, we form the clusters by optimizing the k-means algorithm to better reach our goal. To avoid defective cases of k-means, we propose a new optimized version of this algorithm. For that we precede the k-means classic algorithm by Linde-Buzo-Gray (LBG) [15] algorithm in order to better place centroids (Prototypes) on the plan. The main idea of this algorithm is to start with a small number of prototypes, then copy and adapt according to the distribution of objects in order to increase accuracy and to balance cluster sizes. After that, when the centroids are well positioned we apply the algorithm of the classic k-means[1], while checking that there are no classes with large or small sizes. If the algorithm detects a mega-class it splits it in two, creating a new centroid. If it detects a micro-class, objects are reallocated by removing their centroids. This process is illustrated in figure 1.

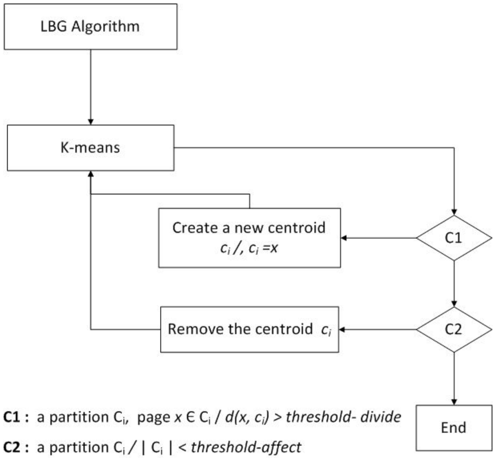

LBG Algorithm
K-means
Create a new centroid
c_i /, c_i =x
C1
Remove the centroid c_i
C2
End
C1 : a partition C_i, page $x \in C_i$ / $d(x, c_i)$ > *threshold- divide*
C2 : a partition C_i / | C_i | < *threshold-affect*

Fig. 1. Organigram of optimized k-means algorithm

3.3 The Incremental Module

We use this module in two cases: First, when a web page arrives in the system, the vector is calculated. Then we use the clusters produced by the previous module. The second case consists of positioning a page if it changes.

In this module we used a modified version of ant based clustering algorithm and the K-means algorithm. Since the objects are positioned on the $Z \times Z$ grid by the previous clustering module, it remains to place a new object in the grid. The cluster model need to be updated periodically as the database changes through insertions and modifications. Suppose we have a monitor that transfers changed data periodically, places the inserted data objects randomly at empty sites in the $Z \times Z$ plane, and replaces modified data objects from the previous plane. The plane keeps records of the entropy and pheromone, and this information can guide newly laden agents to move toward the previously generated clusters.

Although the basic behavior for incremental clustering is the same as the initial clustering, an agent moves according to entropy instead of moving randomly. When new or modified web pages are positioned on the 2D grid, all we have to do is to update the centroids according to the standard incremental k-means[16].

4 Experimentation Results

In this section we present experimental results of our proposed algorithm and compare it against the incremental k-means approach.

4.1 Data-Set and Setup

For all experiments, a single data-set was used as the primary source of input to the algorithms. This data-set was the Bank-search data-set, whose classes are known. Our data-set consists of 10,000 web documents classified into 10 equally-sized groups each containing 1,000 web documents. Each category was chosen by selecting 4 distinct themes from the real world, namely Banking and Finance, Programming Languages, Science and Sport. The most difficult clustering task using this data-set would be the clustering of the entire 10,000 web documents back into the original 10 groups [17].

Since our data-set consists of 10 groups, we had to choice the required number of documents to make our data-set useful. The choice was to download and archive 100, 200, 500 or 1000 documents per group. We choose to start the clustering with a smaller size and increase it incrementally until we reach 1000 document per group. According to [17], to understand clustering performance on 'semantically close' groups, we use two sets of experiments. One attempts to separate semantically similar groups, and the other concentrats on separating semantically distant groups. Varying the number of groups allowed us to measure the relative performance of our clustering when separating between two, three or five groups for both 'similar' and 'disimilar' groups. This set of data-sets is represented in table 1.

Table 1. Testsets setup

TestSet	groups	Num Documents	Num Clusters
D1	Commercial Banks, Building Societies, insurance Agencies	200 To 3000	3
D2	Commercial Banks, Java, Motor Sport	200 To 3000	3
D3	Commercial Banks,insurance Agencies,C,Astronomy,Soccer	200 To 5000	5

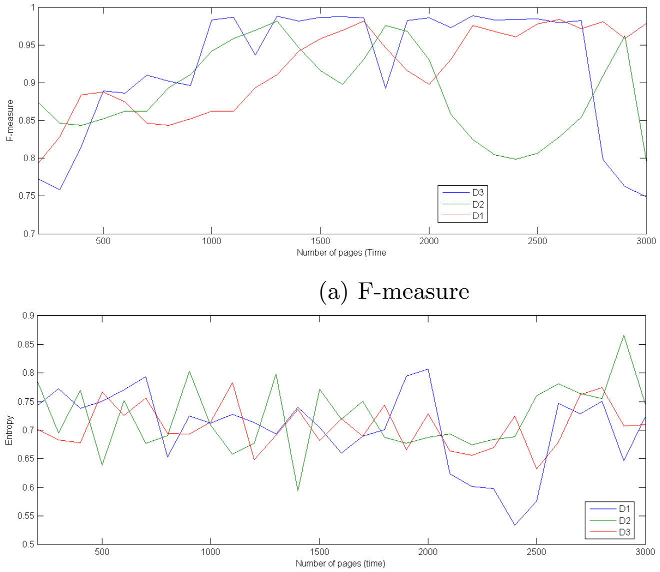
4.2 Clustering Evaluation Measures

Entropy and F-Measure are two main methods widely used for numerically scoring the cluster quality. F-Measure is the weighted harmonic mean of precision and recall and it is often used to measure clustering quality. The higher the value of F-Measure, the better the clustering quality has been got. Entropy is a measure of the randomness of molecules in a thermodynamics system. Entropy is often used to evaluate the clusters distribution of clustering. If documents are distributed uniformly and there are little differences between clusters, the value of Entropy will be high. On the contrary, if there are great differences between clusters, the value of Entropy will be low.

Since in our case the real clustering results are available, we can, besides these measures,compare our results with the pre-classified categories of the bank-search data-set.

4.3 Evaluation Results

In order to evaluate IKACC algorithm, we compare its results with those of the Incremental K-means Clustering (IKC) algorithm using F-Measure and Entropy through incremental web documents arrival. Before that, we have empirically evaluated the method on the different test-sets, that are shown in figure 2. We can see that our method can handle both semantically similar and distant documents, because the cluster's quality has not changed much between D1 and D2. We noticed too that the algorithm evaluated the exact number of clusters through many tests every time.

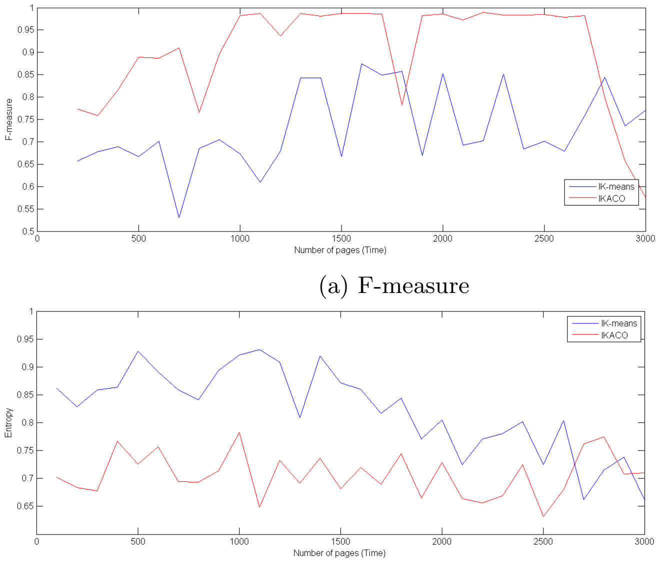


(a) F-measure

(b) Entropy

Fig. 2. clustering quality results on the testsets

Then we compare the clustering quality of IKACC and Incremental K-means. In this group of experiments, we used the D3 testset. It is true that IKACC is slower than the classic IK-means in average of 12 percent. That is because, in every step of incremental clustering, IKACC saves the pheromones, entropy information about every pixel (x, y) in the $Z \times Z$ plane, to guide ants in the next step, while k-means saves only information about centroids. But figure 3 shows that IKACC achieves improvement over IK-means, 19 percent in average for F-Measure and 17 percent in average for Entropy.



(a) F-measure

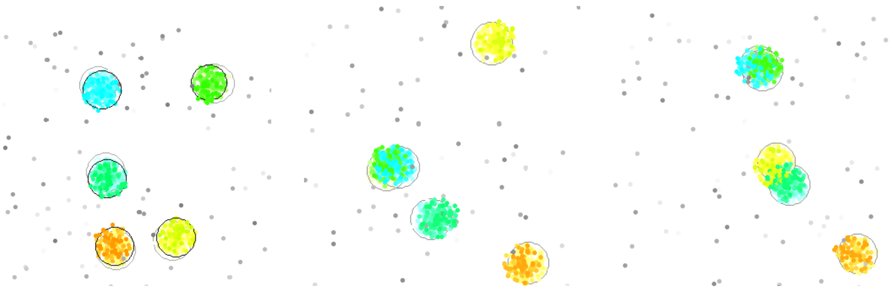
(b) Entropy

Fig. 3. cluster quality comparison between IKACC and IKC

In order to evaluate the efficiency of ant colony clustering step and the use of the LBG algorithm, in our approach, to discover clusters, we picked up some examples of the $Z \times Z$ plan at multiple times of the clustering while processing. We have chosen the parameters as follows (Z=60, s=5, t_{max}=15, N_a=50, λ=0.1 , N_0=200), the test-set as the D2 test-set, where the real number of clusters is K=3. The plans are shown in figure 4. We can clearly see that ants are able to detect the number of clusters, and group data incrementally.

5 Discussion

Compared to non-ant based incremental methods, our approach does not need to pre-specify the number of clusters. It can find clusters boundaries without



(a) First iteration of IKACC N=200, K=5 (b) After 50 iterations N=500, K=4 (c) After 120 iterations N=3000, K=3

Fig. 4. Cluster's formation on the $Z \times Z$ plan

predefined bias. In addition, the factor influencing an agent's picking up or dropping action is entropy: each action can reduce the entropy of the previous patch, and thus speed up clustering. Also, the number of parameters which need to be specified for constructing a clustering model is small. In our approach there are only 6 parameters (Z, s, t_{max}, N_a, λ, N_0). After the clustering, we can compute and save the entropy and pheromone for each pixel in the $Z \times Z$ plane. When each batch of pages arrives, agent movements are guided by the pheromone for locating new objects. Furthermore, the temporal complexity of locating objects in initial clustering is $O(t_{max} \times N_a) + O(N_0)$, and the temporal complexity of computing the entropy and pheromone is $O(Z \times Z)$, it don't depend on the entire number of pages. This model is also able to maintain the quality of the results over time by verifying the coherence of clusters and detecting super clusters and micro clusters through the LBG pre-processing of k-means. Finally, the clusters boundaries are well formed using the spherical shape of k-means.

6 Conclusion

In this paper, we proposed a hybrid clustering algorithm called Incremental K Ant Colony Clustering (IKACC) for web page clustering. The proposed method performs in three modules including the feature extraction of web pages, the clustering and the incremental module. For the two last modules we proposed a hybridization of the ant based clustering and k-means algorithms. The optimized k-means algorithm is used to define the boundaries of clusters; we proposed to use the LBG algorithm for better placing the centroids on the plane given as a result of the ant clustering step. Then the incremental step of our proposition is defined to deal with dynamic data using a modified version of both ant based clustering and k-means clustering.

IKACC outperforms both on effectiveness and efficiency, since we had 92 percent comparing with the real classification of data. Some future issues need to be addressed such as: using different similarity measures than using entropy, and using the IKACC on real data. Our approach can be used for many other applications.

References

1. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proc. Fifth Berkeley Symp. on Math. Statist. and Prob., vol. 1, pp. 281–297. Univ. of Calif. Press (1967)
2. Saatchi, S., Hung, C.-C.: Hybridization of the ant colony optimization with the K-means algorithm for clustering. In: Kalviainen, H., Parkkinen, J., Kaarna, A. (eds.) SCIA 2005. LNCS, vol. 3540, pp. 511–520. Springer, Heidelberg (2005)
3. Wong, W.C., Fu, A.W.C.: Incremental Document Clustering for Web Page Classification. In: IEEE Int. Conference on Society in the 21st Century: Emerging Technologies and New Challenges (IS 2000), Japan (2000)
4. Gavin, S., Yue, X.: Enhancing an incremental clustering algorithm for web page collections. In: 2009 IEEEWICACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, vol. 3, pp. 81–84 (2009)
5. Liu, B., Pan, J., McKay, R.I.B.: Entropy-based metrics in swarm clustering. International Journal of Intelligent Systems 24, 989–1011 (2009)
6. Deneubourg, J.L., Goss, S., Franks, N., Sendova-Franks, A., Detrain, C., Chretien, L.: The dynamics of collective sorting robot-like ants and ant-like robots. In: Proceedings of the First International Conference on Simulation of Adaptive Behavior on from Animals to Animats (1990)
7. Monmarche, N., Slimane, M., Venturini, G.: On Improving Clustering in Numerical Databases With Artificial Ants. In: Floreano, D., Mondada, F. (eds.) ECAL 1999. LNCS, vol. 1674, pp. 626–635. Springer, Heidelberg (1999)
8. Kao, Y., Lee, S.Y.: Combining k-means and particle swarm optimization for dynamic data clustering problems. In: IEEE International Conference on Intelligent Computing and Intelligent Systems, ICIS 2009, vol. 1, pp. 757–761 (2009)
9. Kuo, R.J., Wang, M.J., Huang, T.W.: An application of particle swarm optimization algorithm to clustering analysis. Soft Computing 15, 533–542 (2009)
10. Shu-Chuan Chu, J.F.R.: A clustering algorithm using tabu search approach with simulated annealing for vector quantization. Chinese Journal of Electronics 12, 349–353 (2003)
11. Shang, G., Zaiyue, Z., Xiaoru, Z., Cungen, C.: A new hybrid ant colony algorithm for clustering problem. In: International Workshop on Education Technology and Training and 2008 International Workshop on Geoscience and Remote Sensing, ETT and GRS 2008, vol. 1, pp. 645–648 (2008)
12. Kao, Y.T., Zahara, E., Kao, I.W.: A hybridized approach to data clustering. Expert Systems with Applications 34, 1754–1762 (2008)
13. Youssef, S.M.: A new hybrid evolutionary-based data clustering using fuzzy particle swarm optimization. In: 23rd IEEE International Conference on Tools with Artificial Intelligence 1082-3409/11 (2011)
14. Wang, C., Lu, J., Zhang, G.: Mining key information of web pages: A method and its application. Expert Systems with Applications 33, 425–433 (2007)

15. Linde, Y., Buzo, A.G.R.: An algorithm for vector quantizer design. IEEE Transactions on Communications 28, 84–95 (1980)
16. Chakraborty, S., Nagwani, N.K.: Analysis and study of incremental K-means clustering algorithm. In: Mantri, A., Nandi, S., Kumar, G., Kumar, S. (eds.) HPAGC 2011. CCIS, vol. 169, pp. 338–341. Springer, Heidelberg (2011)
17. Sinkaa, M., Corneb, D.W.: The banksearch web document dataset: investigating unsupervised clustering and category similarity. Journal of Network and Computer Applications 28, 129–146 (2004)

A Qualitative Evaluation of Random Forest Feature Learning

Adelina Tang and Joan Tack Foong

Sunway University, Dept. of Computer Science & Networked Systems, No. 5 Jalan Universiti, Bandar Sunway, 46150 Petaling Jaya, Selangor Darul Ehsan, Malaysia
adelina.tang@ieee.org, 12058590@imail.sunway.edu.my

Abstract. Feature learning is a hot trend in the machine learning community now. Using a random forest in feature learning is a relatively unexplored area compared to its application in classification and regression. This paper aims to show the characteristics of the features learned by a random forest and its connections with other methods.

1 Introduction

1.1 Problem and Motivation

Feature learning has been a hot trend in the machine learning community. It is mainly due to the success of deep learning in traditional machine learning tasks [1] and real world application such as MAVIS (Microsoft Audio Video Indexing Service) [2]. Deep learning itself is the attempt to construct multiple layers of feature representation in such a way that higher level abstractions can be represented.

A feature contributes enormously to the success of machine learning task because it is the input of machine learning algorithms and the only thing they see. Features used to be hand engineered by domain experts to reflect their knowledge of the critical aspects about a particular problem. However, as the problem becomes more complicated, we hope that the machine can take the role of the domain experts and be able to extract most relevant features from the raw data.

1.2 Random Forest as Feature Learning Technique

In this paper, we are going to explore feature learning using random forests [3]. A random forest is an ensemble method that gives good results in classification and regression. However the random forest itself is a much richer structure than can be merely used in these two settings. Criminisi gives a nice overview of using random forest in density estimation, manifold learning, and semi-supervised learning [4].

This paper focuses on the feature learning aspect of the random forest. By analysing the reconstruction of the original data using the learned future, we hope to gain some insight on how it works. Finally, we will discuss briefly its connection with sparse coding [5] and self-taught learning [6].

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_34, © Springer International Publishing Switzerland 2014

2 Literature Review

Deep Learning and Representation Learning. Deep learning is the attempt of learning multiple layers of representation, where the higher level representation is the composition of its lower level counterparts [7].

The first breakthrough of deep learning is the success of deep belief nets [8] in the MNIST [9] digit recognition problem. The state-of-the-art result was long held by the Support Vector Machine (SVM). A more recent breakthrough is achieved in the ImageNet dataset, which achieves 15.3% error rate, lower than the state-of-the-art 26.1% [1]. MAVIS (Microsoft Audio Video Indexing Service) speech system, released in 2012, is based on deep learning [2] as well.

Traditional deep learning has been focusing on various type of neural network such as deep belief net [8], autoencoder [10, 11], Restricted Boltzmann Machine [12], and sparse coding [5, 13]. However, as observed in [7], the ensemble of trees such as boosted trees and random forests can be viewed as a three-level deep architecture. What interests us is not that the ensemble serves as a classifier, but that the outputs from all the trees in the ensemble form a distributed representation [14, 15] of the training data. As the exact form of the representation will be spelled out explicitly in the later part of this article, it suffices now to note that the representation provides a very rich description of the input data in the sense that the number of output patterns it can discriminate is exponential to the number of its parameters [16].

Despite all the good properties mentioned above, only two papers are dedicated to this effort [17, 18]. It is the intention of this article to further investigate the properties of this representation and its application in classification.

Ensemble of Decision Trees. Leo Breiman published his seminal book "Classification and Regression Trees (CART)" [19] in 1993, in which he described the fundamental principles in using decision trees for both classification and regression and paved the way for future research. In the same year, JR Quinlan published one of the most popular tree constructing algorithms "C4.5" in his book "C4.5: Programs for machine learning" [20].

Ensemble methods are ways to combine various weak learners in order to get better result. The idea of combining the strengths of many decision trees is not new. Amit and Geman introduced the use of random generated node tests in constructing many decision trees for handwritten digit recognition in their papers [21, 22] published in 1994 and 1997. The term "Random Decision Forest" was introduced by Ho in his paper [23], in which he used the random partition of the feature space to build the trees.

However, the random forest[1] only began to gain serious attention after Leo Breiman published his seminal paper [3] in 2001. He laid the theoretical framework for random forest and introduced a new way of constructing the decision trees by combing his earlier work in "bagging" [24] and Ho's method.

Random forests and their variants enjoy much success in the fields of machine learning, computer vision and medical imaging [25–29]. In this paper, however,

[1] Random forest is the trademark of Leo Breiman.

we are going to explore the potential of using the random forest as a feature learning algorithm.

3 Methodology

3.1 The Basic of Decision Trees

Decision tree can be regarded as the *partitions of the feature space*. Each node in the decision tree ask a question about the features. The feature space is then split into regions which have distinct answers to the question. Fig. 1 illustrates the splitting process.

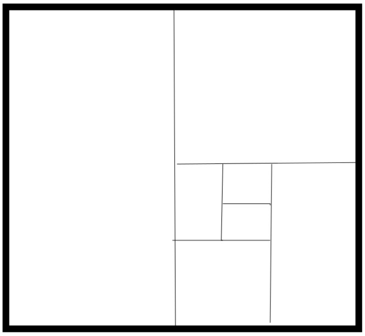


Fig. 1. The decision tree gives rise to the partition of the feature space

3.2 Interpretation of the Partitions

In the common setting of machine learning task, the input data is of the form $[x_i]_{i=1}^n$, where $x_i = (x_i^{(1)}, x_i^{(2)}, ..., x_i^{(m)}) \in \mathbb{R}^m$ is a vector of real number. x_i is reffered as the *data point*, and its components,$(x_i^{(1)}, x_i^{(2)}, ..., x_i^{(m)}$, are referred as *features*. Each node in a decision tree asks a question about the feature, and each distinct answer splits the feature space into corresponding subspaces. Thus each partition in the feature space, and hence each terminal node corresponds to a different configuration and combination of the features. If we consider a feature of a data point as a property that characterizes the data point, then any combination of features can also be regarded as a feature, albeit, a high level feature. Certainly, this high level feature cannot be represented as a real number. However, we can abstract away the detail which is the exact configuration of low level features that correspond to the high level feature, and simply assign each high level feature a terminal node or a distinct symbol. In other words, the decision tree is able to transform the representation of the data point from its standard form $(x_i^{(1)}, x_i^{(2)}, ..., x_i^{(m)})$ to a symbol, say, d (see Fig.2).

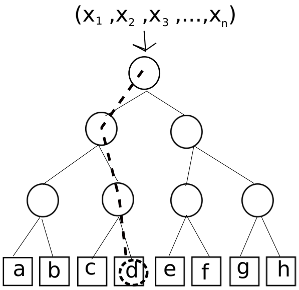


Fig. 2. This diagram shows how a data point is transformed into a symbol

The induced representation of x by $\mathcal{F}$, $\mathcal{F}(x)$ is defined as:

$$\sum_{i=1}^{n}\sum_{j=1}^{m} a_{ij}T_i^{(j)}$$

where $a_{ij} = 1$ if T_i assign the node $T_i^{(j)}$ and $a_{ij} = 0$ otherwise. Note that the summation is purely formal, after all the nodes of decision trees cannot be added, at least not in the usual way. It might just as well be written as a normal vector (a_{ij}). It will be clear in a later section why we choose this notation over a conventional one.

To show that the notation is useful, we use it to introduce an important concept introduced by Breiman: *proximity* [3]. First we have to define a "norm"[2] $||$ for a formal summation of the form $w = \sum_{i=1}^{n}\sum_{j=1}^{m} a_{ij}T_i^{(j)}$ as $|w| = \frac{|a_{ij}|}{n}$.

Given a random forest $\mathcal{F} = \{T_i\}_{i=1}^{n}$ and two data points x and y, the proximity of these two points with respect to $\mathcal{F}$ is the number of identical symbols between the data points divided by n.

Now suppose

$$\mathcal{F}(x) = \sum_{i=1}^{n}\sum_{j=1}^{m} a_{ij}T_i^{(j)}, \mathcal{F}(y) = \sum_{i=1}^{n}\sum_{j=1}^{m} b_{ij}T_i^{(j)}$$

then

$$\begin{aligned} |\mathcal{F}(x) - \mathcal{F}(y)| &= \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}(a_{ij} - b_{ij})T_i^{(j)}}{n} \\ &= \frac{\text{number of different symbols}}{n} \\ &= 1 - \frac{\text{number of identical symbols}}{n} \\ &= 1 - \text{proximity of x and y} \end{aligned}$$

The derivation above shows the natural connection between the "norm" that we defined and the concept of proximity.

[2] Not a norm in the strict mathematical sense.

3.3 Reconstruction of Image

In this section, we are going to show how to reconstruct a binary image from the features induced by the random forest. Suppose an image is represented by a vector of its pixel intensities, i. Given a random forest $\mathcal{F}$, the image can be represented as $\sum_{i=1}^{n}\sum_{j=1}^{m} a_{ij}T_i^{(j)}$ as shown above. In this case, $T_i^{(j)}$ tells us, partially, which pixel is on and which pixel is off. Thus $T_i^{(j)}$ can be regarded as a vector which captures a certain property of the original image. Now the formal sum above can actually be calculated, and the value will be the reconstructed image.

3.4 Tree Building Algorithm

In this paper, we follow closely the algorithm known as *Extremely Randomized Forest* [30]. First of all, a feature,x_i, and a threshold,θ, are chosen randomly. Then the feature space is split into two parts, i.e. $x_i \leq \theta$ and $x_i \geq \theta$. A *score* for this particular split is then calculated. If the score is greater than a predetermined value, repeat the process on the subspaces. although there are many ways to calculate the score of a particular split, the one we are using here is the information gain.

4 Results

The result in this section shows the general properties of the learned representation using the MNIST dataset [9].

Fig. 3 shows the first 15 digits from the dataset.

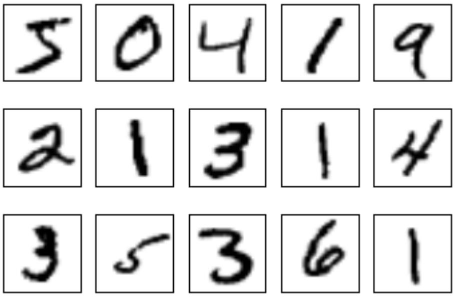


Fig. 3. First 15 digits from the train dataset

A random forest consisting of 30 trees is trained using randomly generated data. To be precise, the data used here consists of 50,000 vectors of dimension 784 × 1, drawn from random uniform distribution. There is no relationship at all with the MNIST dataset. However, it is possible to use this random forest to transform the MNIST dataset into a new representation. The left diagram in Fig. 4 shows the reconstruction of the first 15 digits using the new representation.

For comparison, another random forest is trained using 50,000 digits from the dataset. However, unlike the case of using the random forest in classification, a random label is given for each digit. As shown by the comparison in Fig. 4, the reconstructed digits are more visually recognizable.

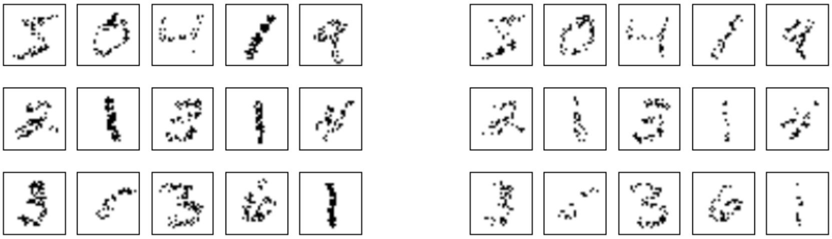


Fig. 4. The left diagram shows the reconstruction using the MNIST dataset, and the right reconstruction using random data

To show the individual contribution of the trees inside the random forest, here is the progressive reconstruction of the digit "5". The diagram is to be read from left to right and from top down. The first image shows the reconstruction using only the first tree; the second image uses the first and second; and the final one uses all of the trees.

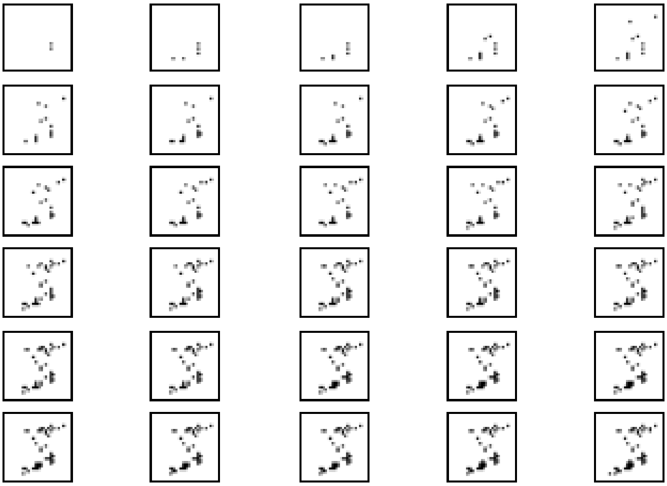


Fig. 5. Progressive reconstruction using random forest trained with random data

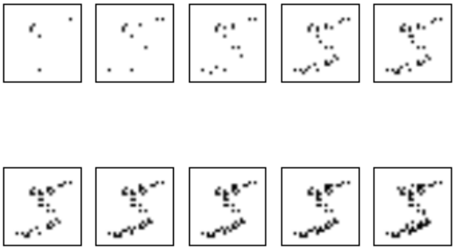


Fig. 6. Progressive reconstruction using random forest trained with original dataset

5 Discussion

In [31], the authors show empirically that the power of sparse coding , as a feature learning technique, is not the learned basis functions but rather the non-linear coding scheme. It corresponds to the facts showed in the paper that the data used to train the random forest is not that important. Instead of justifying our claim using classification accuracy, we choose to reconstruct the images using learned representations. The visual similarity between original images and reconstructed images gives us better intuition.

As shown in Fig. 2, each data point falls to a terminal node through a series of split nodes. Each split node dictates the pixel value of a particular point. Thus a terminal node represents a certain configuration of the pixels. The typical configuration is shown in the top leftmost image in Figs. 5 and 6. It could be just a few points arranged in a particular order, but as they layer up on each other, the digit take its shape gradually (see Figs. 5 and 6).

Take note that the random forest $\mathcal{F}$ can be trained on one set of data X, and yet it can be used in constructing the representation of the data point from another set of data Y. X and Y can have no relation at all, with the exception that their data points must have the same dimensions. In fact, X can be totally random data. As shown in the comparison in Fig. 4, the random forest trained on random data can nevertheless represent the basic shapes of the digits as well as the random forest trained on the digits dataset. The notable difference here is that the pixel density is lower for the digits reconstructed using the random forest trained on random data. The idea of training a learner, using different data from the one on which it eventually applies, is explored in the paper [6], in which the authors coined the term *self-taught learning* as an alternative to the other learning paradigms such as supervised learning, unsupervised learning, transfer learning, and reinforcement learning.

Motivated by this observation, we propose another interpretation of the formal summation[3] $\sum a_{ij}T_i^{(j)}$. Given that data point x is in the form of $(x^{(i)})_{i=1}^n$ and the $T_i^{(j)}$ specifies the values of a certain subset of the features, say $(x^{(k_i)})_{i=1}^m$,

[3] Abbreviated form of $\sum_{i=1}^n \sum_{j=1}^m a_{ij}T_i^{(j)}$.

then $T_i^{(j)}$ can be represented as $(v_i)_{i=1}^n$ where $v_i = x^{(k_j)}$ if $i = k_j$ else $v_i = 0$. We can now say $\sum a_{ij} T_i^{(j)}$ approximates the data point x, that is $x \approx \sum a_{ij} T_i^{(j)}$. Observe that most of the a_{ij} that is zero for each data point is assigned with a single terminal node, and in general there are 2^d terminal nodes for a binary tree with depth d. Suppose n trees in a forest have the same depth, then the ratio of non-zero coefficients in the sum $\sum a_{ij} T_i^{(j)}$ is

$$\frac{(n \times 1)}{(n \times 2^d)} = \frac{1}{2^d} \to 0 \text{ as } d \to \infty$$

In other words, the representation induced by the random forest is very *sparse*. On the other hand, sparse coding [5] is the method of representing a data point x in the form of $\sum a_i T_i$ that minimizes

1. the difference $|x - \sum a_i T_i|$
2. the sum of coefficients $|\sum a_i|$

The second constraint encourages the coefficients to have as many zeros as possible, thus the term *sparse*. Notice the similarity between sparse coding and the method outlined in this paper although the method here achieves sparsity, but not by direct optimization.

The possibility of using a different dataset in the training phase is a plus point for this method. Hence, we can use a combination of a large number of seemingly unrelated datasets to train the learner and then apply the learner to yet another dataset of interest. The more data that we can feed into a learner, the better its performance. The method in this paper can exploit existing patterns as it can learn from unrelated datasets.

In conclusion, the method outlined here shows basic learning capacity and shares a lot of interesting connections with other methods too. The future work will be focused on the elaboration of the connections as well as the application of the learned feature(s) in the classification task.

References

1. Krizhevsky, A., Sutskever, I., Hinton, G.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, vol. 25, pp. 1106–1114 (2012)
2. Seide, F., Li, G., Yu, D.: Conversational speech transcription using context-dependent deep neural networks. In: Proc. Interspeech, vol. 11, pp. 437–440 (2011)
3. Breiman, L.: Random forests. Machine Learning 45(1), 5–32 (2001)
4. Criminisi, A.: Decision Forests: A Unified Framework for Classification, Regression, Density Estimation, Manifold Learning and Semi-Supervised Learning. Foundations and Trends in Computer Graphics and Vision 7(2-3), 81–227 (2011)
5. Lee, H., et al.: Efficient sparse coding algorithms. In: Advances in Neural Information Processing Systems, pp. 801–808 (2006)
6. Raina, R., et al.: Self-taught learning: Transfer learning from unlabeled data. In: Proceedings of the Twenty-fourth International Conference on Machine Learning (2007)

7. Bengio, Y.: Learning deep architectures for AI. Foundations and Trends in Machine Learning 2(1), 1–127 (2009)
8. Hinton, G.E., Osindero, S., Teh, Y.W.: A fast learning algorithm for deep belief nets. Neural Computation 18(7), 1527–1554 (2006)
9. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition Proceedings of the IEEE, Vol. Proceedings of the IEEE 86(11), 2278–2324 (1998)
10. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning internal representations by error propagation. Tech. rep. DTIC Document (1985)
11. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science 313(5786), 504–507 (2006)
12. Salakhutdinov, R., Mnih, A., Hinton, G.: Restricted Boltzmann machines for collaborative filtering. In: Proceedings of the 24th International Conference on Machine Learning, pp. 791–798. ACM (2007)
13. Olshausen, B.A., Field, D.J.: Sparse coding with an overcomplete basis set: A strategy employed by V1? Vision Research 37(23), 3311–3325 (1997)
14. Hinton, G.E.: Learning distributed representations of concepts. In: Proceedings of the Eighth Annual Conference of the Cognitive Science Society, Amherst, MA, pp. 1–12 (1986)
15. Bengio, Y., et al.: Neural probabilistic language models. In: Holmes, D.E., Jain, L.C. (eds.) Innovations in Machine Learning. STUDFUZZ, vol. 194, pp. 137–186. Springer, Heidelberg (2006)
16. Bengio, Y., Delalleau, O., Simard, C.: Decision trees do not generalize to new variations. Computational Intelligence 26(4), 449–467 (2010)
17. Moosmann, F., Nowak, E., Jurie, F.: Randomized clustering forests for image classification. Pattern Analysis and Machine Intelligence 30(9), 1632–1646 (2008)
18. Vens, C., Costa, F.: Random Forest Based Feature Induction. In: 2011 IEEE 11th International Conference on Data Mining (ICDM), pp. 744–753. IEEE (2011)
19. Breiman, L.: Classification and regression trees. CRC Press (1993)
20. Quinlan, J.R.: C4. 5: Programs for machine learning, vol. 1. Morgan Kaufmann (1993)
21. Amit, Y., Geman, D.: Randomized Inquiries About Shape: An Application to Handwritten Digit Recognition. Tech. rep. DTIC Document (1994)
22. Amit, Y., Geman, D.: Shape quantization and recognition with randomized trees. Neural Computation 9(7), 1545–1588 (1997)
23. Ho, T.K.: Random decision forests. In: Proceedings of the Third International Conference on Document Analysis and Recognition, vol. 1, pp. 278–282. IEEE (1995)
24. Breiman, L.: Bagging predictors. Machine Learning 24(2), 123–140 (1996)
25. Bosch, A., Zisserman, A., Muoz, X.: Image classification using random forests and ferns. In: IEEE 11th International Conference on Computer Vision, ICCV 2007, pp. 1–8. IEEE (2007)
26. Criminisi, A., Shotton, J., Robertson, D., Konukoglu, E.: Regression forests for efficient anatomy detection and localization in CT studies. In: Menze, B., Langs, G., Tu, Z., Criminisi, A., et al. (eds.) MICCAI 2010. LNCS, vol. 6533, pp. 106–117. Springer, Heidelberg (2011)
27. Fanelli, G., Gall, J., Van Gool, L.: Real time head pose estimation with random regression forests. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 617–624. IEEE (2011)

28. Geremia, E., Menze, B.H., Clatz, O., Konukoglu, E., Criminisi, A., Ayache, N.: Spatial decision forests for MS lesion segmentation in multi-channel MR images. In: Jiang, T., Navab, N., Pluim, J.P.W., Viergever, M.A., et al. (eds.) MICCAI 2010, Part I. LNCS, vol. 6361, pp. 111–118. Springer, Heidelberg (2010)
29. Leistner, C., et al.: Semi-supervised random forests. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 506–513. IEEE (2009)
30. Geurts, P., Ernst, D., Wehenkel, L.: Extremely randomized trees. Machine Learning 63(1), 3–42 (2006)
31. Coates, A., Ng, A.: The Importance of Encoding Versus Training with Sparse Coding and Vector Quantization. In: Proceedings of the 28th International Conference on Machine Learning, pp. 921–928 (2011)

A Semantic Content-Based Forum Recommender System Architecture Based on Content-Based Filtering and Latent Semantic Analysis

Naji Ahmad Albatayneh*, Khairil Imran Ghauth, and Fang-Fang Chua

Faculty of Computing and Informatics, Multimedia University, Persiaran Multimedia,
63000 Cyberjaya, Selangor, Malaysia
naji.albatayneh@gmail.com,
{khairil-imran,ffchua}@mmu.edu.my

Abstract. The rapidly increasing popularity of social computing has encouraged Internet users to interact with online discussion forums to discuss various topics. Online discussion forums have been used as a medium for collaborative learning that supports knowledge sharing and information exchanging between users. One of the serious problems of such environments is high volume of shared data that causes a difficulty for users to locate relevant content to their preferences. In this paper, we propose an architecture of a forum recommender system that recommends relevant post messages to users based on content-based filtering and latent semantic analysis which in turn will increase the dynamism of online forums, help users to discover relevant post messages, and prevent them from redundant post messages as well as bad content post messages.

Keywords: Recommender system, Content-based filtering, Latent Semantic Analysis.

1 Introduction

The emergence of social computing which is supported by Web 2.0 has attracted internet users to interact via social networking sites and online discussion forums to discuss about various topics . Since so, online discussion forums have been providing new formats for creative expression to be an innovative medium for modern collaborative learning that supports knowledge sharing and information exchanging between users of different experiences and backgrounds. This has led users to face a huge amount of shared information over online discussion forums that might be not interesting or even redundant what caused a difficulty for them to locate and discover interesting post messages. Furthermore, users face difficulty to distinguish between a good post message and a bad post message since most of the online forums normally are not equipped with a rating system as it will pose an extra task to the user to

* Corresponding author.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_35, © Springer International Publishing Switzerland 2014

evaluate and rate the posted message. These were the main reasons of the increasingly interest in recommender systems that have become an important research area since the appearance of the first papers on collaborative filtering in the mid-1990s [2].

Since then, researches have gone off into various directions in the recommender system research area where some researchers in this area have been researching the application of recommender systems in specific domains while other researchers continued working on the algorithmic aspects of recommender systems, yet others focused on the aspects of the user interface of recommender systems. However, despite all of these advances in recommender systems research area, the current generation of recommender systems still requires further improvements to make recommending methods more effective and accurate while the main purpose of recommender systems is still the same which is recommending relevant items to the users using three main filtering techniques which are content-based filtering which is based on item profile, collaborative filtering which is based on user profile, and hybrid filtering which is a combination of both techniques [1], [2].

In this paper, we propose a novel semantic content-based recommender system for online discussion forum based on content-based filtering to recommend similar post messages that will in turn help to prevent users from bad content post messages and reduce the redundancy problem. In addition, we use latent semantic analysis to extract and form the keywords from the post messages.

The rest of this paper is organized as follows. We refer to related work in recommender system domain in general and specifically in the area of forum content-based recommender system in Section 2. In Section 3 we introduce our proposed content-based recommender system framework as well as we formalize the calculations of keywords extraction and formulation from a post message using latent semantic analysis, and the calculations of post messages similarities using vector space model. Section 4 presents a prototype with a brief of an online discussion forum that is integrated with the proposed recommender system. Section 5 concludes and shows the main directions of our future work.

2 Related Works

Recommender systems have emerged as an important response to the so-called information overload problem [6]. Generally, recommender system filters and recommends relevant items to users based on either user profile, item profile or the combination of both. Thus, recommendation techniques are often classified based on the recommendation approach into several categories: content-based filtering, collaborative filtering and hybrid filtering approaches [1], [2]. Recommender systems that are based on content-based filtering are mostly used to recommend similar items of textual form based on the items that the user has liked in the past. Similarity among items is calculated by analyzing keywords overlaps between two items. Among the popular methods to calculate the item similarity in content-based recommender systems are *Vector Space Model* (VSM) [9] and *k-Nearest Neighbors* (kNN). In order to determine the item similarity using VSM method, the angle between vectors represents the similarity percentage where each vector represents a particular item while in kNN method, item similarity is determined by calculating the distance between two points where each point represents a particular item. In contrast to content-based recommender systems, collaborative recommender systems make use of only past user activities (for example, transaction history or user satisfaction expressed in ratings given to items) to calculate user similarity [10]. Among the popular methods to calculate user similarity is *Pearson Correlation Coefficient* which compares a set of ratings given by a pair of users to determine the user similarity [8]. Hybrid recommender systems combine both content-based filtering and collaborative filtering techniques in various ways to solve some limitations in content-based and collaborative recommender systems. This can be done by combining the rating prediction, adding characteristics of collaborative filtering technique into content-based filtering technique, adding characteristics of content-based filtering technique into collaborative filtering technique, and creating a single unifying model that incorporates both techniques' characteristics [2].

Recommender systems have been used in various applications that offers a lot of services or items, one of these applications is online discussion forums where users face a huge amount of shared information. The purposes of such a social environment is diverse and may include exchanging information or social support, or even supporting learning [3]. Several recommender systems techniques have been used for online discussion forums, one of these is [4] which presents and discusses a framework of a rule-based recommender system for online discussion forums while [5] presents and discusses a framework of a semantic VSM-based recommender system for online discussion forums, where most of recommender systems use content-based filtering technique to provide recommendations in online discussion forums due to content-based recommender systems are able to filter and recommend items of textual form.

Most of content-based forum recommender systems lack of discovering the latent association among the terms that have the same or very similar meaning but have different names and thus treat these terms differently which so-called Synonymy problem. For example, the seemingly different terms "children" and "kids" actually have very similar meaning, but content-based forum recommender systems would find no matching between these two terms what decreases recommendation performance. Previous attempts to solve the synonymy problem depended on intellectual or automatic term expansion, or the construction of a thesaurus [11]. Fully automatic methods were inefficient to solve Synonymy problem due to some added terms by automatic methods may have different meanings from intended ones which leads to a serious degradation of recommendation performance.

The Singular Value Decomposition (SVD) technique that is used in Latent Semantic Analysis (LSA) method, is capable of discover the latent association among synonyms where it takes a huge rectangular matrix of terms that are associated with documents and then constructs a semantic space where terms and documents that are closely associated are placed closely to each other. SVD arranges the space to reflect the major associative patterns among terms in the document, and discard the less important ones which helps to deal the Synonyms problem.

3 Proposed Method

As shown in figures 2 and 3, the architecture of the proposed content-based recommender system is based on three main phases of processes which are: 1) Removing stop words from the post messages and stemming the words after retrieving them from the database, 2) Extracting and forming keywords from the post messages using Latent Semantic Mapping, 3) Calculate the similarity among post messages using Vector Space Model which involves two main stages of processes; creating term weights using Term Frequency/Inverse Document Frequency (TF-IDF) and then calculating similarity values between post messages by using Cosine similarity measurement. These phases will be presented and formalized in the following subsections:

3.1 Remove Stop Words, and Words Stemming Using Porter's Algorithm

In this phase all the posted messages will be automatically fetched from the database to be reconstructed. For this purpose, at first symbols, illegal characters, spaces, punctuations, common words in forums (*e.g. thread, quote, reply, ...*), and stop words (*e.g. we, they, are, have*) will be removed from the post messages. Then all of the words will be converted to lowercase mode, after that the high reputation words will be eliminated. Finally, the remained words of post messages will be structured and stored into an array, afterward the Porter's algorithm and Snowball framework [7] will be executed on the array to stem the words (e.g. this set of words will be stemmed to convert them to their roots; {*connection, connections, connective, connected, connecting*} => {*connect*} which is the root of all similar words in the set).

3.2 Extracting and Forming Keywords from Post Messages Using LSM

This process starts with creation of a term by sentences matrix A = $[A_1, A_2, \ldots, A_n]$ with each column vector A_i representing the weighted term frequency vector of sentence i in a post message under consideration. If there are a total of m terms and n sentences in a post message, then we will have an $m \times n$ rectangular matrix A for the post message. Since every word does not normally appear in each sentence, the matrix A is sparse.

Given an $m \times n$ rectangular matrix A, where without loss of generality $m \geq n$, it can be decomposed into the product of three other matrices using the Singular Value Decomposition (SVD) as defined in the following formula (1):

$$A = U \Sigma V^T \tag{1}$$

Where $U = [u_{ij}]$ is an $m \times n$ column-orthonormal matrix whose columns are called left singular vectors; $\Sigma = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$ is an $n \times n$ diagonal matrix, whose diagonal elements are non-negative singular values sorted in descending order, and $V = [v_{ij}]$ is an $n \times n$ orthonormal matrix, whose rows are called right singular vectors, as shown in figure 1.

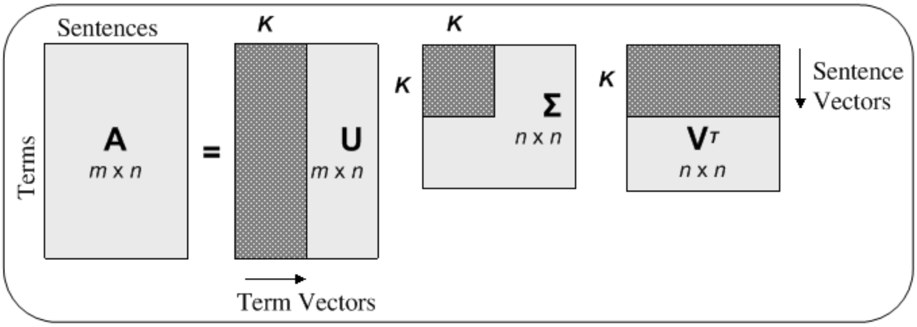


Fig. 1. The Singular Value Decomposition (SVD) of an *m* × *n* rectangular matrix A

We used a truncated SVD which means keeping only the *k* column vectors of U and *k* row vectors of V^T corresponding to the *k* largest singular values Σ*k* while the rest of the matrix is discarded, as shown in figure 1. The purpose of using truncated SVD is to make it easier and more economical to process the approximate matrix as well as quicker than dealing with original matrix since the approximate matrix is in a very useful sense the closest approximation to A that can be achieved by a matrix of rank *k*. The following formula shows that (2):

$$A = U_k \Sigma_k V_k^T \quad (2)$$

3.3 Calculate the Similarity Values among Post Messages Using VSM

Vector space model involves two stages which are creating Term Weights and Cosine Similarity. In term weights calculation process, the post messages will be analyzed and constructed into vectors of keywords which will be used to calculate term weight W*i,j* by using one of the best measurements of the keywords weights which is the Term Frequency/Inverse Document Frequency (TF-IDF). The following formula (3) calculates term weight of term *i* in post message *j* (W*i,j*) using TF-IDF.

$$w_{i,j} = \frac{f_{i,j}}{\max_z f_{z,j}} * \log\left(\frac{D}{d_i}\right) \quad (3)$$

Where $f_{i,j}$ denotes the frequency of term *i* occurring in post message *j*, $max_z f_{z,j}$ is the maximum frequency among all the *z* keywords that appear in post message *j*, *D* is the total number of post messages that can be recommended to a user, and d_i is the number of post messages that contain term *i*. After calculating all terms weights of all terms of a post message, the weight vector for post message *j* will be as following:

$$Vj = [\ W_{1,j}\,,\ W_{2,j}\,,\ W_{3,j}\,,\ \ldots\,,\ W_{N,j}\,] \quad (4)$$

The second stage of calculations in VSM is calculating the similarity value between two post messages by using term weights vectors as shown in formula (4) in the cosine similarity formula (5). The relevancy rankings of the post messages are measured based on the deviation angles between two post messages vectors, similarity can be calculated using *cosine* similarity as follows:

$$SIM = \cos\left(\overrightarrow{w_c}, \overrightarrow{w_s}\right) = \frac{\overrightarrow{w_c} \cdot \overrightarrow{w_s}}{\left\|\overrightarrow{w_c}\right\| \left\|\overrightarrow{w_s}\right\|} = \frac{\sum_{i=1}^{n} w_{c,i} \cdot w_{s,i}}{\sqrt{\sum_{i=1}^{n} w^2_{c,i}} \sqrt{\sum_{i=1}^{n} w^2_{s,i}}} \tag{5}$$

Where $\overrightarrow{w_c}$ represents a vector of post message *c* and $\overrightarrow{w_s}$ represents a vector of post message *s*. Both $\left\|\overrightarrow{w_c}\right\|$ and $\left\|\overrightarrow{w_s}\right\|$ are the magnitude of the vectors $\overrightarrow{w_c}$ and $\overrightarrow{w_s}$.

In vector space model if a post message is newly added, deleted or even its attribute updated, the similarity value must be recalculated again. For this purpose, AJAX technology will be used in the proposed recommender system to keep the similarity values between post messages up to date. The proposed content-based

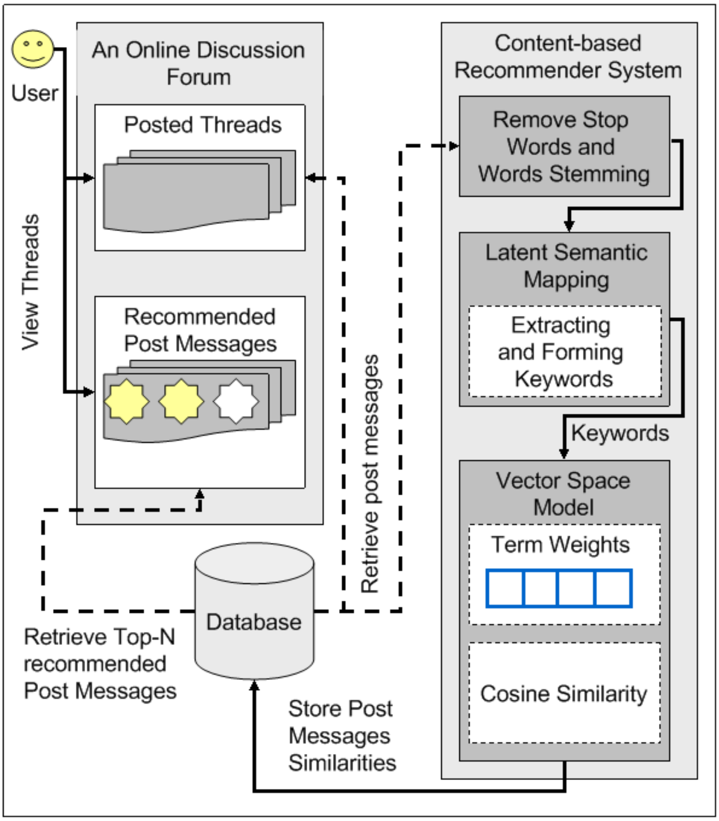

User
View Threads
An Online Discussion Forum
Posted Threads
Recommended Post Messages
Retrieve post messages
Retrieve Top-N recommended Post Messages
Database
Store Post Messages Similarities
Content-based Recommender System
Remove Stop Words and Words Stemming
Latent Semantic Mapping
Extracting and Forming Keywords
Keywords
Vector Space Model
Term Weights
Cosine Similarity

Fig. 2. The system architecture of the proposed content-based recommender system

recommender system calculates similarity between query post messages (the post messages that the user has liked or rated high in the past) and the other post messages that are posted by other users in the forum, then all similar post messages will be sorted based on the similarity value and then stored in the database. After that, the proposed recommender system recommends top-N similar post messages to the user as shown in figures 2 and 3.

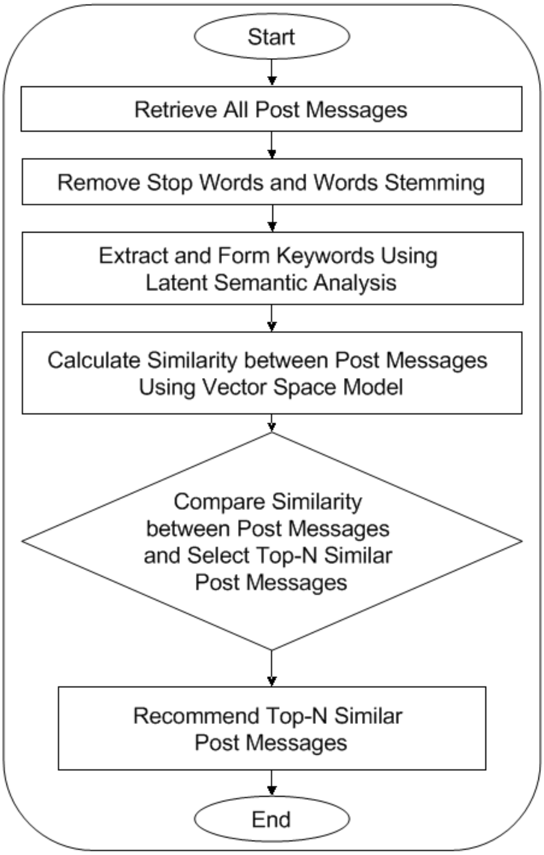
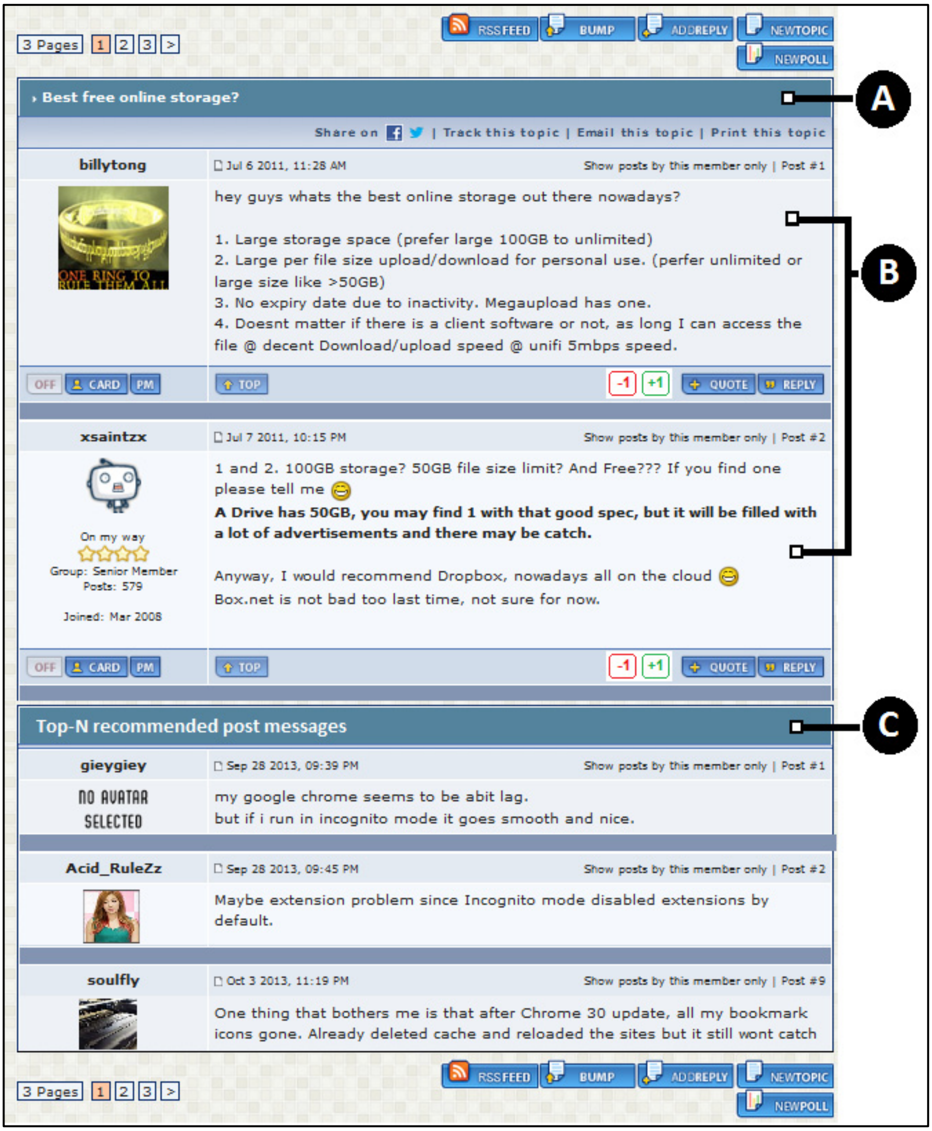

Start
Retrieve All Post Messages
Remove Stop Words and Words Stemming
Extract and Form Keywords Using Latent Semantic Analysis
Calculate Similarity between Post Messages Using Vector Space Model
Compare Similarity between Post Messages and Select Top-N Similar Post Messages
Recommend Top-N Similar Post Messages
End

Fig. 3. A flowchart of the main processes of the proposed recommender system

4 Prototype

Figure 4 shows a screen shot of an online discussion forum that is integrated with the proposed recommender system. Label (A) shows the first post message in the current viewing thread by the user, this post message is considered as a title of the current thread. Label (B) shows the reply messages that discuss the first post message in the current thread, these reply messages which are posted by several users could be



› Best free online storage?

billytong

Jul 6 2011, 11:28 AM — Show posts by this member only | Post #1

hey guys whats the best online storage out there nowadays?

1. Large storage space (prefer large 100GB to unlimited)
2. Large per file size upload/download for personal use. (perfer unlimited or large size like >50GB)
3. No expiry date due to inactivity. Megaupload has one.
4. Doesnt matter if there is a client software or not, as long I can access the file @ decent Download/upload speed @ unifi 5mbps speed.

xsaintzx

On my way
Group: Senior Member
Posts: 579
Joined: Mar 2008

Jul 7 2011, 10:15 PM — Show posts by this member only | Post #2

1 and 2. 100GB storage? 50GB file size limit? And Free??? If you find one please tell me

A Drive has 50GB, you may find 1 with that good spec, but it will be filled with a lot of advertisements and there may be catch.

Anyway, I would recommend Dropbox, nowadays all on the cloud
Box.net is not bad too last time, not sure for now.

Top-N recommended post messages

gieygiey

Sep 28 2013, 09:39 PM — Show posts by this member only | Post #1

my google chrome seems to be abit lag.
but if i run in incognito mode it goes smooth and nice.

Acid_RuleZz

Sep 28 2013, 09:45 PM — Show posts by this member only | Post #2

Maybe extension problem since Incognito mode disabled extensions by default.

soulfly

Oct 3 2013, 11:19 PM — Show posts by this member only | Post #9

One thing that bothers me is that after Chrome 30 update, all my bookmark icons gone. Already deleted cache and reloaded the sites but it still wont catch

Fig. 4. Screen-shot of Online discussion forum— label [C] shows a sorted list of top-N recommended post messages to the users

relevant to the topic of the thread or not, they also might contain good content or even bad content. Forum's users can like or dislike these reply messages to send these ratings to the recommender system where they will be analysed. Label (C) shows a sorted list of top-N recommended post messages to the current user by the proposed content-based recommender system, these recommended post messages are similar to the user's preference, contain good content and posted by other trusted users. Bad content or bad structured post messages will be discarded even if they are similar to the user's preference. Moreover, the post messages that their similarity values are higher than 90% will be discarded as well to prevent users from redundant post messages or alternatives.

5 Conclusion and Future Work

In this paper, we have presented and discussed a novel architecture of a semantic content-based recommender system for online discussion forums based on content-based filtering and latent semantic analysis where we used vector space model to calculate the similarity between post messages in two stages of calculations which are: 1) Terms weights calculations by one of the best measurements of keywords weights which is the Term Frequency/Inverse Document Frequency (TF-IDF). 2) Cosine similarity calculation which is a measure of similarity between two vectors of keywords. Moreover, we used latent semantic mapping to extract and form keywords from post messages as well as to identify keywords from post messages since a post message may contain too many words that have potential to become keywords. Our long-term goal is the design of an architecture of a content-based recommender system that is more accurate and reliable as well as compatible with any type of online discussion forums to recommend accurate and trusted recommendations even for new users. So one of the future work can be considering the contextual data of users for accuracy improvement. Moreover, future work will include testing of the current proposed architecture of the content-based recommender system and comparing it with the other existing recommender systems in terms of accuracy, trust and performance.

References

1. Adomavicius, G., Manouselis, N., Kwon, Y.: Multi-criteria recommender systems, pp. 769–803. Springer Science+Business Media, LLC (2011)
2. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering, 734–749 (2005)
3. Webster, A., Vassileva, J.: Visualizing personal relations in online communities. In: Wade, V.P., Ashman, H., Smyth, B. (eds.) AH 2006. LNCS, vol. 4018, pp. 223–233. Springer, Heidelberg (2006)
4. Abel, F., Bittencourt, I.I., Henze, N., Krause, D., Vassileva, J.: A Rule-Based Recommender System for Online Discussion Forums. In: Nejdl, W., Kay, J., Pu, P., Herder, E. (eds.) AH 2008. LNCS, vol. 5149, pp. 12–21. Springer, Heidelberg (2008)
5. Hadi, F.T., Mehran, Y.: A Semantic VSM-Based Recommender System. International Journal of Computer Theory and Engineering 5(2) (2013)
6. John, O., Barry, S.: Trust in recommender systems. In: Proceedings of the 10th International Conference on Intelligent User Interfaces, pp. 167–174 (2005)
7. Llia, S.: Overview of Stemming Algorithms. Mechanical Translation (2008)

8. Liu, N.N., Qiang, Y.: EigenRank: a ranking-oriented approach to collaborative filtering. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 83–90 (2008)
9. Pasquale, L., Marco de, G., Giovanni, S.: Content-based Recommender Systems: State of the Art and Trends. Recommender Systems Handbook, pp. 73–105. Springer (2011)
10. Takács, G., Pilászy, L., Németh, B., Tikk, D.: Scalable Collaborative Filtering Approaches for Large Recommender Systems. The Journal of Machine Learning Research, 623–656 (2009)
11. Xiaoyuan, S., Taghi, M.K.: A Survey of Collaborative Filtering Techniques. Advances in Artificial Intelligence 2009 (2009)

A Simplified Malaysian Vehicle Plate Number Recognition

Abd Kadir Mahamad[1,2], Sharifah Saon[2], and Sarah Nurul Oyun Abdul Aziz[2]

[1] Embedded Computing System (EmbCoS),
[2] Faculty of Electrical and Electronic Engineering
Universiti Tun Hussein Onn Malaysia
86400 Parit Raja, Batu Pahat,
Johor, Malaysia
kadir@uthm.edu.my

Abstract. This paper propose an automatic inspection system of alphabets and numbers to recognize Malaysian vehicles plate number based on digital image processing and Optical Character Recognition (OCR). An intelligent OCR Training Interface has been used as a library and the system has been developed using LabVIEW Software. This software then is used to test with different situation to ensure the proposed system can be applied for real implementation. Based on the results, the proposed system shows good performance for inspection and can recognize an alphabets and numbers of vehicle plate number. To sum up, the proposed system can recognize the alphabets and numbers of the Malaysian vehicles plate number for inspection.

Keywords: LabVIEW, OCR, Adobe Photoshop, Plate Number.

1 Introduction

The inspection and pattern recognition processes have been performed by human to verify the object or character based on the image by eye is not always correct. Hence, inspection using human visual is inefficient, increases the problem during processing the image and cannot be able to give 100% of accurate results for inspection process. Considering the competitive nature and high expectation from user, this inspection technique must be change to automated inspection system.

Digital image processing methods is one of the processes that developed with combination computer technology and camera [1]. Initially, image processing method required the pattern recognition process for one shapes of object or images. Objects or images recognition process is useful for human to analyze a character that have in color or gray- level character [2][3].

The objective of this project is to develop the Malaysian vehicles plate number recognition system using LabVIEW 2012, using OCR Training Interface and validate the system with different situation on plate number. To achieve the objectives, five (5) sample of vehicles plate number is trained. Adobe Photoshop software is used to make the 5 sample image is look like real and make it with 10 different of situation

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014,* Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_36, © Springer International Publishing Switzerland 2014

for each sample. The process begins by detecting the vehicle plate number with the perfect recognition process, status of inspection, the owner data of vehicles and the result of features for monitoring.

2 Literature Review

Image processing is used to describe operations which carried out on images, with the aim of accomplishing some functions. The process is to convert the image into a form where it can be more easily transmitted over a telecommunication link or stored in computer memory. It might also can reduce the noise or to extract information of particular interest to a human observer. Common image processing operations include pre-processing, character detection, character segmented and character recognition [4].

2.1 Optical Character Recognition

OCR is normally used to inspect the application of an identification or classification of the component, automatically. The proposed system should able to detect, identify and differentiate various types of text on various types of surface. Through this system, it can identify and detect series of Malaysian vehicle's plate number while the vehicle is moving. By using OCR, the system can be developed where it can help to identify each character quickly along with the appropriate process [5].

2.2 Vision Assistant

Vision Assistant through National Instrument (NI) is software package that focuses on making the prototype of vision applications. It also helps the user to understand machine vision and image processing functions better [6].

3 Research Methodology

The proposed system of this project consists of two main modules: (1) License plate locating Region of Interest (ROI). (2) Vehicle plate number identification module using Optical Character Recognition (OCR) to recognize individual character for each sample which consists of alphabets and numbers. The proposed flowchart of the system is mentioned in Fig. 1.

The proposed system reads an input image taken by the camera and passes it to the pre-processing unit. From there, the process continue by passing the image to vehicle plate detection unit then the image will be sent to OCR process [6][7]. If the vehicle number is recognize, it will be compared with the database and displayed through GUI of LabVIEW.

3.1 Image Acquisition

The input of this system (image) is captured by using a camera with a distance of 2 to 3 meters away from the vehicles as shown in Fig. 2. The picture resolution can be set manually by 640 × 480 pixels. In this study, 55 pictures are considered (normal and different conditions) and compiled into a single video file. This will be easier for the system to read each images automatically.

3.2 Pre-processing Input Image

The RGB image is then converted into gray scale image for easy analysis as it consists of only two colour channels [7][8]. Tools as in Fig. 3, IMAQ VI Create in LabVIEW is used to convert the image input from a video file from 32-bit to 8-bit to make it easier to read the character by the system.

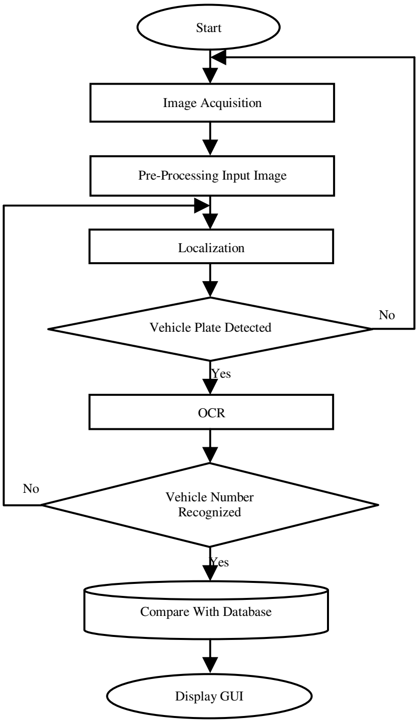


Fig. 1. The Proposed System Workflow

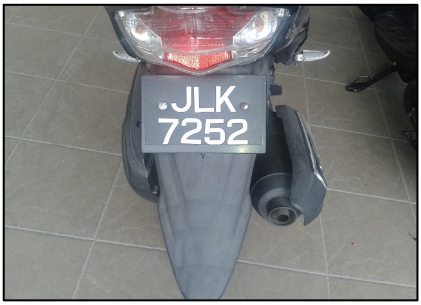


Fig. 2. Original Image for Recognition

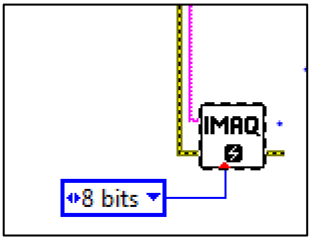
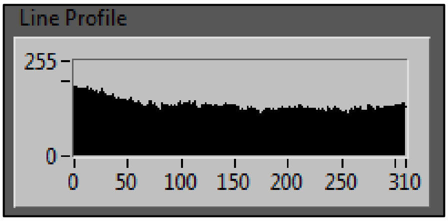
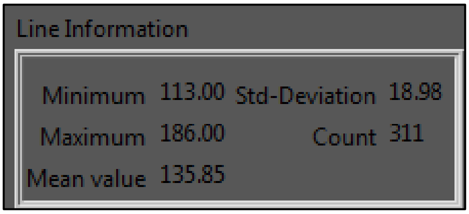


Fig. 3. IMAQ IV Tools

Grayscale image calculates the intensity of light and it contains 8 bits (or one Byte or 256 values i.e. $2^8 = 256$). Each pixel in the grayscale image represents one of the 256 values, in particular the value 0 represents black, 255 represents the white and the remaining values represents intermediate shades between black and white [4]. Gray level of all pixels is scaled into the range of 0 to 311 and compared with the original range 0 to 255, the character pixels. Fig. 4 shows the graph of Line Pixel and Fig. 5 shows the total of minimum pixel, maximum pixel, mean and standard deviation of vehicle plate number that acquired from Line Profile.




Line Information

Minimum 113.00 Std-Deviation 18.98
Maximum 186.00 Count 311
Mean value 135.85

Fig. 4. Line Pixel

Fig. 5. Value of Line Information

3.3 Localization

Localization for vehicle plate recognition module is further divided into the following subtasks.

IMAQOCR Read Text 3 VI. The output tools IMAQ OCR Text 3 VI of ROI descriptor is used to identify the vehicle plates number where the ROI must be a rectangle and the ID part (string condition) externally (shown in Fig. 6).

Identifying of Horizontal and Vertical Edges. The identification method uses vertical and horizontal projection to perform vertical and horizontal segmentation. The identification method analyses the vertical and horizontal of vehicle plate number as shown in Fig. 7.

Extracting Vehicle Plate Number. In this function, we extract the shape of the vehicles plate number from the image including the front or rear view of the vehicle. In Malaysia, size and shape of vehicle plate numbers are various. Including different condition that might accidentally affected the vehicle plate number, such as rain, dazzled and etc. Therefore, we need to extract the shape of vehicles plate number in consideration of these conditions.

3.4 Character Recognition

The most important stage of vehicles plate number is character recognition. At this stage, the character images that are extracted from vehicles plate number have to be recognized [9]. Before recognizing characters, the patterns have to be created. There are several types of patterns: image of each character, different sizes or conditions and others.

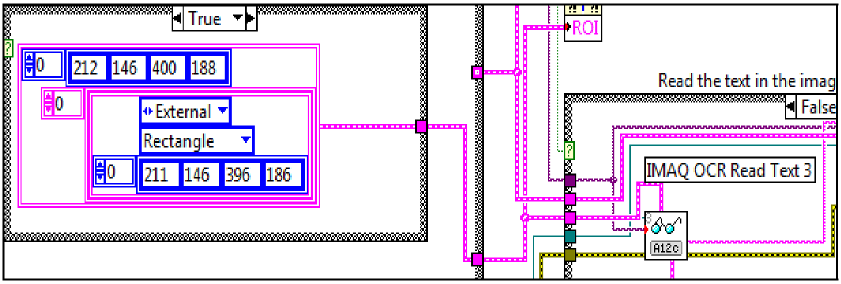
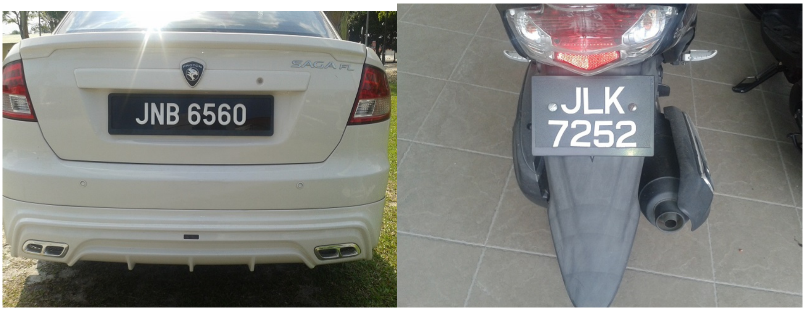


Fig. 6. ROI Descriptor



(a) (b)

Fig. 7. (a) Serial and (b) Parallel Plate Number for ROI Descriptor

Focusing on the Upper and Lower Part of Parallel and Serial Plate Number. The proposed method only works on Malaysian standardized vehicles plate number that consists of parallel and serial pattern. The top and bottom part (parallel) or single part (serial) of the extracted plate number is specified (shown in Fig. 8). Thus, we need to focus on the region of interest, which consists of the character that needs to be identified.

Dilation Operation to Separate the Alphabets and Numbers. The character separation is done by dilation operation where each character is cut off into block of character (shown in Fig. 9). In another word, it is employed to convert an image of text into characters. OCR is an analytical artificial intelligence system that considers sequences of characters rather than whole words or phrases.

Matching Characters by Using the OCR. To link the character set file (numbers and alphabets which been trained earlier and stored in the OCR Training Interface). This process is shown in the circuit below (Fig. 10). When running the LabVIEW program, “IMAQ OCR Read Character Set File” tools is used in order to identify each character. This process is called matching process.

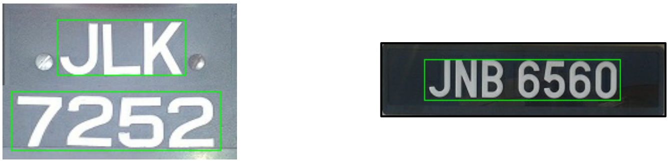


Fig. 8. Upper and Lower Part of Parallel and Serial Plate

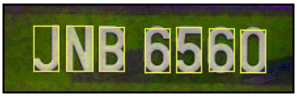


Fig. 9. Recognition of Character in Blocks

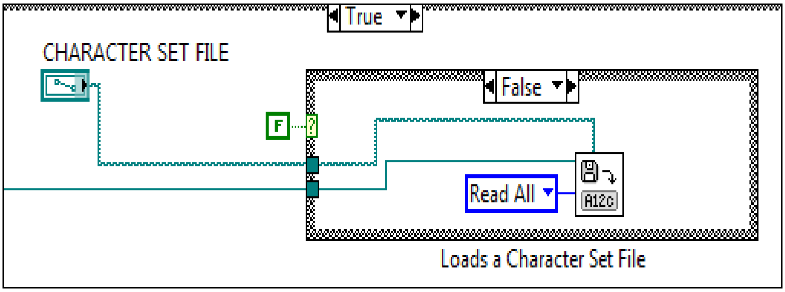


Fig. 10. VI of Linking OCR Character Set File

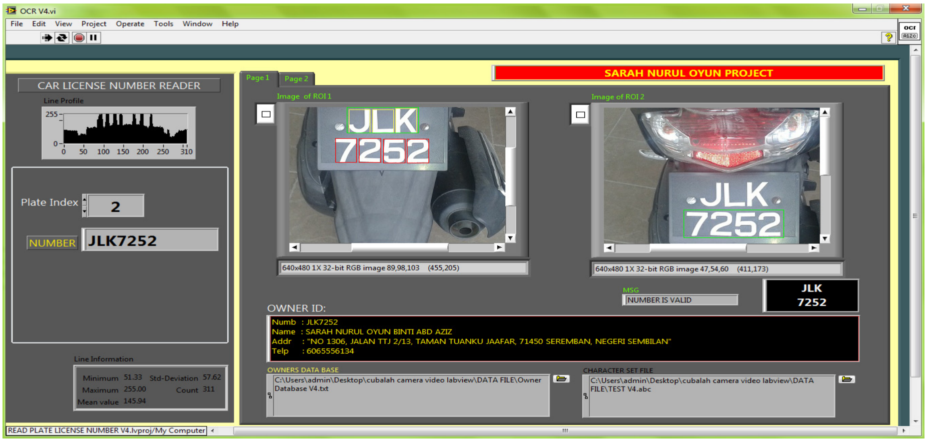
4 Results and Analysis

4.1 Results

In this project, the experiments have been performed in LabVIEW 2012 which on Intel Pentium 1.50GHz PC. These images were taken from:

1. Different situation edited by Adobe Photoshop
2. Various locations like garage, roadside and parking lots.
3. Those images were taken from the length of 2 to 3 meters distance.
4. The size of the input image is 640 x480.

Overall 55 images were considered from various conditions. The proposed system successfully inspects 37 image of Malaysian vehicles plate number. This is because when a vehicle is face with a different situation, the LabVIEW system will be difficult to get a proper reading of the characters OCR. Apart from that, the parameter set to read the characters also can't recognized the alphabet successfully depends on the situation that face on the vehicles plate number. When the system is face with the image is more critical situation, the ROI is unable to detect the character correctly as well as the result. The GUI of the vehicle plate recognition and sample of recognition of plate number in ten conditions are shown in Fig. 11 and 12 respectively.



SARAH NURUL OYUN PROJECT

CAR LICENSE NUMBER READER

Plate Index 2

NUMBER JLK7252

Image of ROI 1

640x480 1X 32-bit RGB image 89,98,103 (455,205)

Image of ROI 2

640x480 1X 32-bit RGB image 47,54,60 (411,173)

MSG NUMBER IS VALID

JLK
7252

OWNER ID:

```
Numb  : JLK7252
Name  : SARAH NURUL OYUN BINTI ABD AZIZ
Addr  : "NO 1306, JALAN TTJ 2/13, TAMAN TUANKU JAAFAR, 71450 SEREMBAN, NEGERI SEMBILAN"
Telp  : 6065556134
```

Line Information

Minimum	51.33	Std-Deviation	57.62
Maximum	255.00	Count	311
Mean value	145.94		

OWNERS DATA BASE
C:\Users\admin\Desktop\cubalah camera video labview\DATA FILE\Owner Database V4.txt

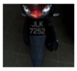
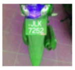
CHARACTER SET FILE
C:\Users\admin\Desktop\cubalah camera video labview\DATA FILE\TEST V4.abc

READ PLATE LICENSE NUMBER V4.lvproj/My Computer

Fig. 11. GUI Interface of the Vehicle Plate Recognition with OCR

Situation	Dazzled	Blur	Mud	Bright	Night	Snow	Rain	Fracture	Dusty	Autumn
Vehicle Plate Number " JNB 6560" In 10 Condition										
Status	Not Invalid	Not Invalid	Not Invalid	Invalid	Not Invalid	Not Invalid	Not Invalid	Invalid	Not Invalid	Invalid
Owner Data	Not Displlay	Not Display	Not Display	Display	Not Display	Not Display	Not Display	Not Display	Not Display	Display

(a)

Situation	Dazzled	Blur	Mud	Bright	Night	Snow	Rain	Fracture	Dusty	Autumn
Vehicle Plate Number "JLK 7252" In 10 Condition										
Status	Invalid	Not Invalid	Invalid	Invalid	Invalid	Invalid	Not Invalid	Invalid	Not Invalid	Invalid
Owner Data	Display	Not Display	Display	Display	Display	Display	Not Display	Display	Not Display	Display

(b)

Fig. 12. Plate Number Recognition with Ten Difference Condition for (a) car and (b) motorcycle

4.2 Analysis

Analysis were conducted to test the proposed system and to measure the accuracy of the system. Average of the recognize value over images is conducted to produce a more reliable and confident result of the proposed system [9], especially toward Malaysian vehicles plate number with the different situation. The complexity of each of these subsections of the program is determines the accuracy of the overall system.

Accuracy Based on Situation of Image. The approach used for recognized the vehicle plate number is using OCR and LabVIEW with different situation on each sample of Malaysia vehicle plate number. The testing set has been chosen properly and it is representative for the problem existing that may occur during inspection. Each time a character is submitted to the LabVIEW software it is able to recognize the character successfully.

Analysis based on Table 1 shown that bright and autumn situation is compatible used with this system because the parameter set from OCR easy to inspect the vehicle plate number [9]. The situation of blur and dusty gave a lowest accuracy because the LabVIEW system reject a character whose recognition is not sufficiently reliable with parameter set using OCR.

The problems encountered in the earlier systems in order to locate the number plate when plate numbers have different situations were overcome by morphological operation which achieving higher accuracy in number plate extraction step. As the fonts vary from one number plate to the other, ambiguous situation may arise in recognizing the characters ‘G’ and ’0’, ‘I’ and ‘1’, ‘7’ and ‘T’ and alike since OCR template was developed for one particular font. But some of them were overcome by “character categorization” approach.

Table 1. Accuracy Based On Situation of Image

Situation Image	Total Image Tested	Plate Detection Accuracy %
Dazzled	5	60%
Blur	5	20%
Mud	5	80%
Bright	5	100%
Night	5	60%
Snow	5	40%
Rain	5	40%
Fracture	5	40%
Dusty	5	20%
Autumn	5	100%

Effect of OCR Acceptance Level on Reading Accuracy of Vehicle Plate Number. Sample of plate number consists of 55 images of vehicle plate number with different situation indicates acceptance level in OCR Training Interface is used to match a trained character. The OCR acceptance level is examined from 500 to1000 on reading accuracy varied from 73.84% to 0 %, as shown in Table 2. In order to select the best accuracy is set to the higher than 65%. To make the system inspection to read the correct string of Malaysian vehicles plate number, the accuracy level must higher than 65%.

Table 2. Effect of OCR Acceptance Level on Vehicle Plate Number Reading Accuracy

OCR Acceptance Level	Reading Accuracy
500	73.84%
600	64.61%
700	61.53%
800	50.77%
900	6.15%
1000	0%

5 Conclusion

This paper presented a method for detection and recognition of Malaysian vehicle plate number. The proposed technique of automated plate number recognition is divided into two modules; Vehicles plate Detection by using Region of Interest (ROI) and Optical Character Recognition (OCR). Although the system is customized to handle specific format plate number of a specific country, but it can be used for multinational vehicle plate number especially in detection of the plate number. Since characters and numbers are used for most of the countries vehicle plate number, so OCR technique of recognition of characters is applicable to any similar plate number with the change in the templates stored in the database

Acknowledgment. The financial support received from the Exploratory Research Grant Scheme (ERGS) (Vot No: E036), Office for Research, Innovation, Commercialization and Consultancy Management (ORICC), University Tun Hussein Onn Malaysia is gratefully acknowledged.

References

1. Mohammad, K.: Automatic Text Detection And Digital Character Recognition. International Scholarly Research Network: ISRN Machine Vision (2010)

2. Khalifa, O.: Malaysian Vehicle License Plate Recognition. The International Arab Journal of Information Technology 4(4) (2007)
3. Ibrahim, S.A.: Malaysian License Plate Number Detection Based On Sobel Vertical Edge Algorithm. Universiti Teknologi MARA (2007)
4. Maglad, K., Mohamad, D.: Nureddin: Saudian Car License Plate Number Detection and Recognition. Australian Journal of Basic and Applied Sciences 5(12), 1780–1786 (2011)
5. Gonzalez, R.C., Woods, R.E.: Digital Image Processing, 2nd edn. (2002)
6. Pratt, W.K.: Digital Image Processing, 3rd edn. (2001)
7. Pearson, D.E.: Image Processing. McGraw-Hill, London (1991)
8. Optical Character Recognition., `http://webmastersoftwareprojects4u.com`
9. Jain, A.K.: Fundamentals of Digital Image Processing. Prentice-Hall Inc., Englewood Cliffs (1989)

Agglomerative Hierarchical Co-clustering Based on Bregman Divergence

Guowei Shen, Wu Yang, Wei Wang, Miao Yu, and Guozhong Dong

Information Security Research Center, Harbin Engineering University, Harbin,
Heilongjiang Province, China 150001
{shenguowei,yangwu,w_wei,yumiao,dongguozhong}@hrbeu.edu.cn

Abstract. Recently, co-clustering algorithms are widely used in heterogeneous information networks mining, and the distance metric is still a challenging problem. Bregman divergence is used to measure the distance in traditional co-clustering algorithms, but the hierarchical structure and the feature of the entity itself are not considered. In this paper, an agglomerative hierarchical co-clustering algorithm based on Bregman divergence is proposed to learn hierarchical structure of multiple entities simultaneously. In the aggregation process, the cost of merging two co-clusters is measured by a monotonic Bregman function, integrating heterogeneous relations and features of entities. The robustness of algorithms based on different divergences is tested on synthetic data sets. Experiments on the DBLP data sets show that our algorithm improves the accuracy over existing co-clustering algorithms.

Keywords: Co-clustering, Bregman divergence, Agglomerative hierarchical algorithm, Heterogeneous Information networks.

1 Introduction

Co-clustering is widely used in heterogeneous information networks mining, clustering different entities simultaneously. Current co-clustering algorithms require prior knowledge of cluster number, and the entities divided to obtain their flat structure [1] and [2]. However, accurate estimation of the number of clusters is very difficult, and the flat structure can't reflect the real situation [3]. Therefore, hierarchical co-clustering algorithm is proposed to mine hierarchical structure of entities simultaneously, without requiring any prior knowledge.

How to accurately measure the distance is very difficult in hierarchical co-clustering algorithms. The existing hierarchical co-clustering algorithms only consider a single metric based on prior knowledge [4]. However, distributions of real networks are Poisson, multinomial, or even exponential distribution.

Some Bregman divergences had been proposed to measure the distance in machine learning [2-6], including K-L divergence, I-divergence, squared Euclidean distance, Itakura-Saito distance and so on. K-L divergence is often used in text data analysis, and squared European distance is used in the classical K-means algorithm. Under the

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_37, © Springer International Publishing Switzerland 2014

uniform framework of Bregman divergence metric, we propose an agglomerative hierarchical algorithm to analyze data sets with different statistical distributions.

In the agglomerative process, current algorithms only consider the relationship between entities, while the real network might include outlier points. The integration of entity features can solve the outlier co-clustering problem. Motivated by the merge cost function [6], we propose a co-cluster merge cost function based on Bregman divergence, incorporating entity-self features. The cost function satisfies monotonic and provides a more accurate metric in the agglomeration process.

In summary, the main contributions of this paper are:

1. The integration of heterogeneous relationships and features of entities can solve the outlier co-clustering problem in the agglomerative process.
2. A co-cluster merge cost function based on Bregman divergence is proposed.
3. An agglomerative hierarchical co-clustering algorithm based on merge cost function is proposed to learn hierarchical structure of entities simultaneously.
4. The robustness of divergences and the accuracy of algorithms are empirically analyzed based on synthetic and real data sets.

The rest of the paper is organized as follows: We describe related works in Section 2. Section 3 gives a formal definition of the problem and proposes the merge cost function based on Bregman divergence. We present the detail of algorithm in Section 4. Experiments and discussion are in Section 5. Section 6 draws conclusions.

2 Related Works

The problem of co-clustering has been studied extensively in recent literatures. Dhillon et al proposed a co-clustering algorithm based on the maximization mutual information [1]. The mutual information was generalized to Bregman divergence [2-5], which can be used to measure distance of data with different distributions. However, heterogeneous information network has hierarchical structure, such as log data [7] and music data [3]. Therefore, hierarchical co-clustering algorithms are proposed to divide and agglomerate entities [8].

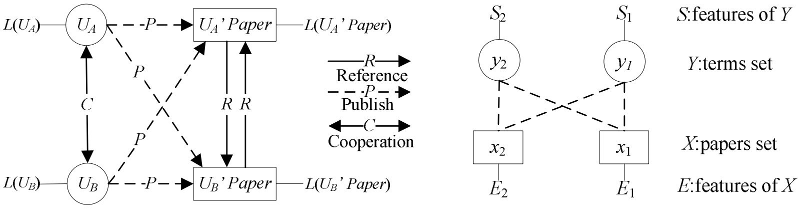
Wei Cheng et al. proposed a hierarchical co-clustering algorithm based on entropy split [4], and binary splits was replaced by N-ary splits in [9]. However, the divided approach is more difficult when dealing with big data, and does not consider features of entities.

Agglomerative hierarchical co-clustering algorithm is widely used in many applications. Tao Li used agglomerative hierarchical co-clustering algorithm to organize the music data [10], but the outlier co-cluster is not considered. Metadata can be used as supervision information to improve the accuracy of co-clustering [11] and [12]. Therefore, the features of entities are integrated into our algorithm.

3 Problem Definition

In this section, we employ academic network data and introduce a hierarchical analysis model.

An academic network includes paper, conference, author, term and other entities, the co-operation relationship among the authors, the reference relationship among the papers and other heterogeneous relationships. Mining the hierarchical structure of paper is a common problem in data mining. Therefore, in this paper, paper and author are the case of these entities shown in Fig. 1(a). $L()$ is feature function of the entity. The heterogeneous network can be modeled as a bipartite graph, shown in Fig. 1(b). In the graph model, *X* represents papers, *E* is features of *X*, *Y* represents terms, and *S* is the features of *Y*.



(a) Academic Network (b) A bipartite graph

Fig. 1. The model of Academic Network

Definition 1: Given a bipartite graph, let X and Y be random variables that take values in the sets $X=\{x_1,x_2\cdots x_m\}$ and $Y=\{y_1,y_2,\cdots y_n\}$ respectively. Under constraints of feature sets *E*, *S*, learning the hierarchical structure $\{(\hat{X}^0,\hat{Y}^0),(\hat{X}^1,\hat{Y}^1),\ldots,(\hat{X}^h,\hat{Y}^h)\}$ from *X* and *Y*. $(\hat{x}_k^h,\hat{y}_l^h)$ represents a co-cluster. Since the hierarchical structures of *X* and *Y* are unbalanced, so usually $k \neq l$.

In this paper, we propose an agglomerative hierarchical co-clustering algorithm based on Bregman divergence to solve the problem. Firstly, co-cluster merge cost function is proposed to measure the distance of two co-clusters.

For a clear description, firstly, we introduce the Bregman divergence [2] and [5]. Let ϕ be a real-valued convex function define on the convex set $\varsigma = dom(\phi) \subset \mathbb{R}$, the Bregman divergence $d_\phi : \varsigma \times \text{int}(\varsigma) \mapsto [0,\infty)$ is defined

$$d_\phi(\boldsymbol{X},\boldsymbol{Y}) = \phi(\boldsymbol{X}) - \phi(\boldsymbol{Y}) - \langle \boldsymbol{X} - \boldsymbol{Y}, \nabla\phi(\boldsymbol{Y})\rangle \quad . \tag{1}$$

where $\nabla\phi$ is the gradient of ϕ.

Motivated by the single side hierarchical clustering based on Bregman divergence [7], we propose a co-cluster merge cost function for agglomerative hierarchical co-clustering.

Definition 2: Given a statistic map τ and distribution metric w, the center of co-cluster $(\hat{x},\hat{y})$ is defined as

$$\tau(\hat{x},\hat{y}) := \sum\nolimits_{x\in\hat{x},\, y\in\hat{y}} w(x,y)\tau(x,y) \quad . \tag{2}$$

Definition 3: Given a co-cluster $(\hat{x},\hat{y})$ and Bregman divergence d_ϕ, the single co-cluster cost is

$$f_{\phi,\tau}(\hat{x},\hat{y}) = \sum_{x\in\hat{x},\, y\in\hat{y}} d_\phi(\tau(x,y),\tau(\hat{x},\hat{y})) \quad . \tag{3}$$

So we can define two co-clusters merge cost as

$$\Delta f_{\phi,\tau}((\hat{x}_1,\hat{y}_1);(\hat{x}_2,\hat{y}_2)) = f_{\phi,\tau}(\hat{x}_1\cup\hat{x}_2,\hat{y}_1\cup\hat{y}_2) - \sum_{j\in\{1,2\}} f_{\phi,\tau}(\hat{x}_j,\hat{y}_j) \quad . \tag{4}$$

Given a convex function ϕ and Bregman divergence d_ϕ, using the definition 2 and 3, two co-clusters merge cost function as

$$\Delta f_{\phi,\tau}((\hat{x}_1,\hat{y}_1);(\hat{x}_2,\hat{y}_2)) = \sum_{i\in\{1,2\}} w(\hat{x}_i,\hat{y}_i) d_\phi(\tau(\hat{x}_i,\hat{y}_i),\tau(\hat{x}_1\cup\hat{x}_2,\hat{y}_1\cup\hat{y}_2)) \quad . \tag{5}$$

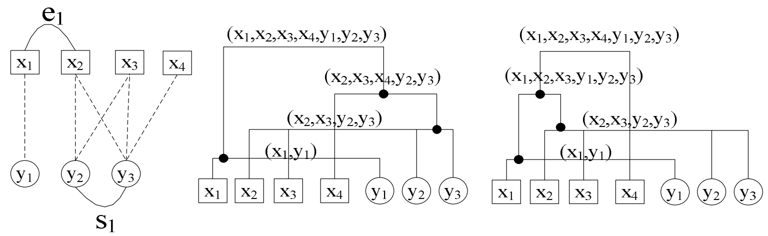
To facilitate understanding, we give two examples. If the divergence is I-divergence, we can exhibit the following derivation. Give random variables Z, $Z \sim w(z)$, $z \in Z \subset \varsigma$ and distribution metrics $w = \{w_{xy} : [x]_1^m, [y]_1^n\}$, so $w_{xy} = p(Z(x,y) = z_{xy})$ .The center of co-cluster $(\hat{x},\hat{y})$ is $\tau(\hat{x},\hat{y}) = E[Z \mid \hat{x},\hat{y}]$. The two co-clusters merge cost is

$$\Delta f_{\phi,\tau}((\hat{x}_1,\hat{y}_1);(\hat{x}_2,\hat{y}_2)) = \sum_{x,y} w_{xy} d_\phi(z_{xy},\hat{z}_{xy}) \quad . \tag{6}$$

If the divergence is squared Euclidean distance, and distribution metrics is $w_{xy} = \{x = x, y = y\}$, so the two co-clusters merge cost is

$$\Delta f_{\phi,\tau}((\hat{x}_1,\hat{y}_1);(\hat{x}_2,\hat{y}_2)) = \frac{|(\hat{x}_1,\hat{y}_1)||(\hat{x}_2,\hat{y}_2)|}{|(\hat{x}_1,\hat{y}_1)| + |(\hat{x}_2,\hat{y}_2)|} \left\|\tau(\hat{x}_1,\hat{y}_1) - \tau(\hat{x}_2,\hat{y}_2)\right\|_2^2 \quad . \tag{7}$$

This merge cost function is generated from the k-means algorithm, which is proposed by Ward [13]. Squared Euclidean distance is used in k-means.



(a) Outlier co-cluster (b) Traditional algorithm (c) Integration of features

Fig. 2. Dendrogram of agglomerative hierarchical co-clustering

An example of outlier co-clustering is shown in Fig. 2(a). Fig. 2(b) shows the dendrogram, which is built by existing algorithms. The outlier co-cluster $(\hat{x}_1, \hat{y}_1)$ can't be handled. If the features of *X* and *Y* are considered, x_1 and x_2 should belong to a same cluster for their same feature e_1 . After integrating the features, the dendrogram is shown in Fig. 2(c). Therefore, we integrate the features $E = \{e_1, e_2, ...e_r\}$ and $S = \{s_1, s_2, ..., s_t\}$ to co-cluster merge cost function.

Through the above example analysis, we can integrate the features into Eq.4. For the feature is not need to be clustered, the cost function is

$$\begin{aligned}&\Delta f_{\phi,\tau}((\hat{x}_1, \hat{y}_1);(\hat{x}_2, \hat{y}_2)) = \sum_{i=\{1,2\}} \sum_{j\{1,2\}} w(\hat{x}_i, \hat{y}_j) d_\phi(\tau(\hat{x}_i, \hat{y}_j) - \tau(\hat{x}, \hat{y}))\\ &+\alpha \sum_{i=\{1,2\}} w(\hat{x}_i, E) d_\phi(\tau(\hat{x}_i, E) - \tau(\hat{x}, E)) + \beta \sum_{i=\{1,2\}} w(S, \hat{y}_i) d_\phi(\tau(S, \hat{y}_i) - \tau(S, \hat{y}))\end{aligned} \quad . (8)$$

Where α, β are the trade-off parameters that balance the effect to the cluster of *X*,*Y*.

Eq. 8 is monotonic and convergent. Due to space constraints, the derivation process is slightly off. Eq. 8 can be solved by iteration of X and Y, decomposed into Eq. 9 and Eq. 10.

$$\begin{aligned}\Delta f_{\phi,\tau}((\hat{x}_1, \hat{y});(\hat{x}_2, \hat{y})) = \arg\min_{\hat{y},E}(&\sum_{i=\{1,2\}} w(\hat{x}_i, \hat{y}) d_\phi(\tau(\hat{x}_i, \hat{y}) - \tau(\hat{x}, \hat{y}))\\ &+ \alpha \sum_{i=\{1,2\}} w(\hat{x}_i, E) d_\phi(\tau(\hat{x}_i, E) - \tau(\hat{x}, E)))\end{aligned} \quad . \quad (9)$$

$$\begin{aligned}\Delta f_{\phi,\tau}((\hat{x}, \hat{y}_1);(\hat{x}, \hat{y}_2)) = \arg\min_{\hat{x},S}(&\sum_{i=\{1,2\}} w(\hat{x}, \hat{y}_i) d_\phi(\tau(\hat{x}, \hat{y}_i) - \tau(\hat{x}, \hat{y}))\\ &+ \beta \sum_{i=\{1,2\}} w(S, \hat{y}_i) d_\phi(\tau(S, \hat{y}_i) - \tau(S, \hat{y})))\end{aligned} \quad . \quad (10)$$

4 Hierarchical Co-clustering Algorithm

In this section, we present details of our hierarchical co-clustering algorithm. Eq.8 can be solved by Eq.9 and Eq.10. So we can develop an agglomerative hierarchical co-clustering algorithm based on minimization of the merge cost function.

The detail of the algorithm BHCC is shown in Algorithm 1. $T(\cdot,\cdot)$ is joint probability used in K-L divergence, $T(\cdot,\cdot)$ is linear mapping in squared Euclidean distance. Step 1-3 initialize the algorithm. The centers of co-clusters are calculated in step 3. In step 5, the function $MergeX()$ merges two co-clusters of *X*, the merge cost of which is minimal. Similarly, the function $MergeY()$ merges two co-clusters of *Y* in step 6.

After merging *X* and *Y*, the new centers of co-cluster $\tau(\hat{x}, \hat{y})$ are calculated in step 8. It should be noted that only new merged co-clusters need to update the center.

During each iteration, the total cost decreases gradually. When $|\hbar|$ is equal to 1, the algorithm terminates.

Algorithm 1. Bregman Hierarchical Co-Clustering (BHCC)

```
Input: $T(X,Y),T(X,E),T(Y,S)$
Output: Hierarchical Co-cluster tree $\{(\hat{X}^0,\hat{Y}^0),(\hat{X}^1,\hat{Y}^1),...,(\hat{X}^h,\hat{Y}^h)\}$
Method:
    Initialize h=0, $\hbar := \{(\hat{X}^0,\hat{Y}^0) \mid X^0 = \{\hat{x}_1,...,\hat{x}_m\}, \hat{Y}^0 = \{\hat{y}_1,...,\hat{y}_n\}\}$
    For each co-cluster calculate the center: $\tau(\hat{x},\hat{y}) = \frac{\sum_{x\in\hat{x}}\sum_{y\in\hat{y}} w(x,y)\tau(x,y)}{\sum_{x\in\hat{x}}\sum_{y\in\hat{y}} w(x,y)}$
    Calculate $T(\hat{X},\hat{Y})$, $T(\hat{X},E)$, $T(\hat{Y},S)$
    While $|\hbar| > 1$ do //(it can be optimized by $\Delta f_X, \Delta f_Y$)
        Merge row cluster:
            $\{(\hat{X}^{h+1},\hat{Y}^h),\Delta f_X\} = MergeX((\hat{X}^h,\hat{Y}^h),T(\hat{X},\hat{Y}),T(\hat{X},E),\tau(\hat{X},\hat{Y}))$
        Merge column cluster:
            $\{(\hat{X}^h,\hat{Y}^{h+1}),\Delta f_Y\} = MergeY((\hat{X}^h,\hat{Y}^h),T(\hat{X},\hat{Y}),T(\hat{Y},S),\tau(\hat{X},\hat{Y}))$
        $f^{h+1} = f^h - \Delta f_X - \Delta f_Y$
        For the new co-cluster update the center:
            $\tau(\hat{x},\hat{y}) = \frac{\sum_{x\in\hat{x}}\sum_{y\in\hat{y}} w(x,y)\tau(x,y)}{\sum_{x\in\hat{x}}\sum_{y\in\hat{y}} w(x,y)}$
        Update $T(\hat{X},\hat{Y}),T(\hat{X},E),T(\hat{Y},S)$ from $(\hat{X}^{h+1},\hat{Y}^{h+1})$
        $\hbar := \{(\hat{X}^h,\hat{Y}^h)\}$ , $h = h+1$
    End while
    Return the Co-cluster tree $\{(\hat{X}^0,\hat{Y}^0),(\hat{X}^1,\hat{Y}^1),...,(\hat{X}^h,\hat{Y}^h)\}$
```

Function 1. $MergeX((\hat{X}^h,\hat{Y}^h),T(\hat{X},\hat{Y}),T(\hat{X},E),\tau(\hat{X},\hat{Y}))$

```
Input: $(\hat{X}^h,\hat{Y}^h)$, $T(\hat{X},\hat{Y})$, $T(\hat{X},E)$, $\tau(\hat{X},\hat{Y})$
Output: $(\hat{X}^{h+1},\hat{Y}^h)$, Minimum Merge cost: $\Delta f$
    For any two co-cluster $(\hat{x}_1,\hat{y}),(\hat{x}_2,\hat{y}) \in (\hat{X}^h,\hat{Y}^h)$
        Calculate $\Delta f_i$ by Eq. 9
    $\Delta f = \arg\min \Delta f_j$
    $(\hat{X}^{h+1},\hat{Y}^h) = (\hat{X}^h,\hat{Y}^h) - (\hat{x}_1,\hat{y}) - (\hat{x}_2,\hat{y}) + (\hat{x}_1 \cup \hat{x}_2,\hat{y})$
    $(\hat{X}^{h+1},E) = (\hat{X}^h,E) - (\hat{x}_1,E) - (\hat{x}_2,E) + (\hat{x}_1 \cup \hat{x}_2,E)$
    Calculate the center of new co-cluster by Eq.11
    Return $(\hat{X}^{h+1},\hat{Y}^h)$, $\Delta f$
```

$$\tau(\hat{x}, E) = \frac{\sum_{x \in \hat{x}} \sum_{y \in E} w(x, y) \tau(x, y)}{\sum_{x \in \hat{x}} \sum_{y \in E} w(x, y)} . \tag{11}$$

The implement of $MergeX()$ is shown in function1. During each iteration, we choose the minimum merge cost of two co-clusters to merge. In step 2, the merge costs are calculated by Eq. 9. The centers of entity and feature are calculated in step 6.

The implement of $MergeY()$ is similar with $MergeX()$, so it is omitted.

5 Experiments

In this section, we perform experiments based on synthetic and real data sets to evaluate the robustness and accuracy of our algorithm. All experiments use F1-measure as co-clustering metrics.

5.1 Data Set

In order to analyze co-clustering algorithm, Lomet et al designed a benchmark data sets [14]. The data sets provide three types of data, which are generated by models with Gaussian, Poisson and multinomial distribution respectively. We use this data sets to analyze the robustness of different divergences.

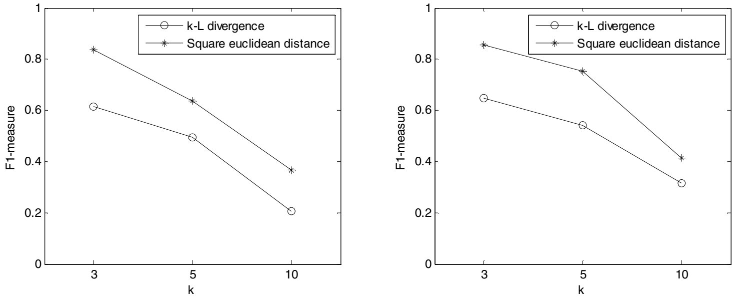
DBLP data set is a classic academic network data set. We choose a sub data set provided by Ming Ji [15] to compare our algorithm with other co-clustering algorithms. The data set includes four areas: database, data mining, information retrieval and artificial intelligence.

The sub data set includes 4057 authors, 100 papers, 20 conferences and 50 high-frequency words. With this data set, papers and authors are represented by *X* and *Y* respectively. conferences and high-frequency words are as features of paper. Thus the constructed *T*(*X*, *Y*) is 100×4057, *T*(*X*, *E*1) is 100×20 and *T*(*X*, *E*2) is 100×50. The feature of paper is conference (denoted DBLP1) or word (denoted DBLP2).

5.2 Robustness Analysis of Different Divergences

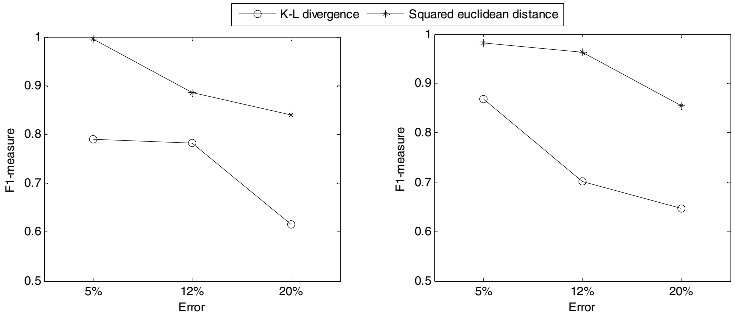
In this section, we analyze the robustness of our algorithm based on different divergences. K-L divergence is tested in data set with Poisson distribution; squared Euclidean distance is tested in data set with Gaussian distribution.

Firstly, we run the algorithm based on different number of cluster *K*, fixing clustering error = 20%, and data size m=n=100. The experiment results are shown in Fig. 3. Comparative results show that the squared Euclidean distance gains higher accuracy. The two kinds of divergence with the value of *K* increases, the clustering accuracy decreases significantly.



(a) row cluster (b) column cluster

Fig. 3. Result of differece divergece based on the number of co-cluster(error=20%, *m*=*n*=100)



(a) row cluster (b) column cluster

Fig. 4. Result of differece divergece based on the difficulty of co-clustering(*K*=3,*m*=*n*=100)

In order to compare the robustness of different divergences based on different difficulties, fixing *m*=*n*=100, *K*=3. Data sets provide the error parameter to control co-clustering difficulty. The larger error, the more difficult co-clustering.

The results are shown in Fig. 4 reveal that the accuracy of co-clustering algorithm decreases with the difficulty increasing. Comparing with the results in Fig. 3, the accuracy of our algorithm decreased slightly while the error increases, so the robustness of our algorithm is better.

5.3 Test on DBLP Data Set

In this experiment, some flat co-clustering algorithm, including ITCC [1], FNMTF [16] and hierarchical co-clustering algorithm HICC [4], are compared with our algorithm. Algorithm FNMTF based on non-negative matrix tri-factorization, which is similar with k-means algorithm, is chosen as a squared European distance divergence contrast algorithms. In reference [4], the results show that the precision of HCC [4] is lower than that of HICC, so we only choose HICC as the comparison algorithm. Our algorithm is divided into two types, BHCC_I based on I-divergence, BHCC_S based on squared Euclidean distance divergence. In this experiment, we set $\alpha = 0.4, \beta = 0$ in our algorithm.

The comparative results of co-clustering algorithms are shown in Fig. 5. Either in flat or hierarchical co-clustering algorithms, our algorithm is more accurate. Since the features are not considered in ITCC, FNMTF and HICC, the results in different data sets DBLP1 and DBLP2 are consistent in the results.

Comparing with algorithms BHCC_S and FNMTF using the same metric, our algorithm BHCC_S has the higher accuracy for the introduction of features, which fully explains the benefits of the introduction of features.

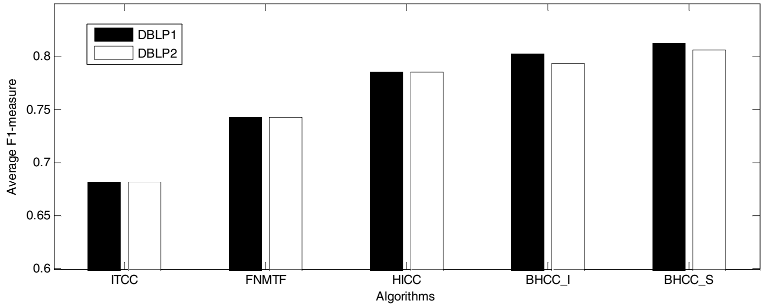


Fig. 5. The comparative results of co-clustering algorithm

The comparative results of BHCC_I and BHCC_S in two different data sets show that our algorithm is more accurate in DBLP1, which takes the conference as the feature of paper. Each paper corresponds to a conference, so T(X, E1) has more information. The outlier co-clusters can be avoided. In DBLP2, since some paper don't include high frequency words, T(X, E2) is very sparse. Therefore, the academic network data mining should select the right feature, which can improve the accuracy of algorithms.

6 Conclusions

In heterogeneous information network mining, hierarchical co-clustering can learn hierarchical structure simultaneously. In this paper, an agglomerative hierarchical co-clustering algorithm based on Bregman divergence is proposed. To solve the problem of outlier co-clusters in the process of co-clustering, the merge cost function not only considers heterogeneous relations, but also integrates the features of entities. Different divergences analyzed on the synthetic data set show that our algorithms are robust. Compared with other flat and hierarchical co-clustering, our algorithm obtained higher accuracy in DBLP datasets.

The present study is based on the divergence determined by priori knowledge of the data set. Our future work is to study the divergence learning from dataset and extensive experiments using other criteria.

Acknowledgments. This work is supported by the National Natural Science Foundation of China (61170242) and 863 Program (2012AA012802).

References

1. Dhillon, I.S., Mallela, S., Modha, D.S.: Information Theoretic co-clustering. In: 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 89–98. ACM, Washington (2003)
2. Banerjee, A., Dhillon, I., Modha, D.S.: A Generalized Maximum Entropy Approach to Bregman Co-clustering and Matrix Approximation. In: 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 509–514. ACM, Washington (2004)
3. Li, J., Shao, B., Li, T., Ogihara, M.: Hierarchical Co-clustering: A New Way to Organize the Music Data. IEEE Transactions on Multimedia 14(2), 471–481 (2012)
4. Cheng, W., Zhang, X., Pan, F., Wang, W.: Hierarchical Co-clustering based on Entropy Splitting. In: 21st ACM International Conference on Information and Knowledge Management, Maui, pp. 1472–1476 (2012)
5. Banerjee, A., Merugu, S., Dhillon, I., Ghosh, J.: Clustering with Bregman Divergence. Journal of Machine Learning Research 6, 1705–1749 (2005)
6. Matus, T., Sanjoy, D.: Agglomerative Bregman Clustering. In: 29th International Conference on Machine Learning, Edinburgh, pp. 1527–1534 (2012)
7. Hosseini, M., Abolhassani, H.: Hierarchical co-clustering for web queries and selected uRLs. In: Benatallah, B., Casati, F., Georgakopoulos, D., Bartolini, C., Sadiq, W., Godart, C. (eds.) WISE 2007. LNCS, vol. 4831, pp. 653–662. Springer, Heidelberg (2007)
8. Mandhani, B., Joshi, S., Kummamuru, K.: A Matrix Density based Algorithm to Hierarchically Co-cluster Documents and Words. In: 12th International Conference on World Wide Web, Budapest, pp. 511–518 (2003)
9. Ienco, D., Pensa, R.G., Meo, R.: Parameter-free hierarchical co-clustering by *n*-ary splits. In: Buntine, W., Grobelnik, M., Mladenić, D., Shawe-Taylor, J. (eds.) ECML PKDD 2009, Part I. LNCS, vol. 5781, pp. 580–595. Springer, Heidelberg (2009)
10. Li, J., Li, T.: HCC: a Hierarchical Co-clustering Algorithm. In: Special Interest Group on Information Retrieval, Geneva, pp. 861–862 (2010)
11. Huang, F., Yang, Y., Li, T., Zhang, J., Rutayisire, T., Mahmood, A.: Semi-supervised Hierarchical Co-clustering. In: Li, T., Nguyen, H.S., Wang, G., Grzymala-Busse, J., Janicki, R., Hassanien, A.E., Yu, H. (eds.) RSKT 2012. LNCS, vol. 7414, pp. 310–319. Springer, Heidelberg (2012)
12. Wu, M.-L., Chang, C.-H., Liu, R.-Z.: Co-clustering with Augmented Data Matrix. In: Cuzzocrea, A., Dayal, U. (eds.) DaWaK 2011. LNCS, vol. 6862, pp. 289–300. Springer, Heidelberg (2011)
13. Ward, J., Hierarchical Grouping, H.: to Optimize an Objective Function. Journal of the American Statistical Association 58(301), 236–244 (1963)
14. Lomet, A., Govaert, G., Grandvalet, Y.: Design of Artificial Data Tables for Co-clustering Analysis. Technical Report, France (2012)
15. Ji, M., Sun, Y., Danilevsky, M., Han, J., Gao, J.: Graph Regularized Transductive Classification on Heterogeneous Information Networks. In: Balcázar, J.L., Bonchi, F., Gionis, A., Sebag, M. (eds.) ECML PKDD 2010, Part I. LNCS, vol. 6321, pp. 570–586. Springer, Heidelberg (2010)
16. Wang, H., Nie, F., Huang, H., Makedon, F.: Fast Nonnegative Matrix Tri-factorization for Large-scale Data Co-clustering. In: International Joint Conference on Artificial Intelligence, Barcelona, pp. 1553–1558 (2011)

Agreement between Crowdsourced Workers and Expert Assessors in Making Relevance Judgment for System Based IR Evaluation

Parnia Samimi and Sri Devi Ravana

Department of Information Systems,
Faculty of Computer Science and Information Technology,
University of Malaya, Malaysia
parniasamimi62@gmail.com, sdevi@um.edu.my

Abstract. Creating a gold standard dataset for relevance judgments in IR evaluation is a pricey and time consuming task. Recently, crowdsourcing, a low cost and fast approach, draws a lot of attention in creating relevance judgments. This study investigates the agreement of the relevance judgments, between crowdsourced workers and human assessors (e.g TREC assessors), validating the use of crowdsourcing for creating relevance judgments. The agreement is calculated for both individual and group agreements through percentage agreement and kappa statistics. The results show a high agreement between crowdsourcing and human assessors in group assessment while the individual agreement is not acceptable. In addition, we investigate how the rank ordering of systems change while replacing human assessors' judgments with crowdsourcing by different evaluation metrics. The conclusion, supported by the results, is that relevance judgments generated through crowdsourcing produces is more reliable systems ranking when it involves measuring of low performing systems.

Keywords: retrieval, evaluation, crowdsourcing, relevance assessment, TREC.

1 Introduction

Gold standard database has an important role in Information Retrieval (IR) evaluation. They are used to compare the quality and effectiveness of systems [1]. Usually, a number of assessors prepare the gold standard dataset which can be reused in later experiments. A common evaluation approach is test collections that also referred to Cranfield experiments which is believed the beginnings of today's laboratory retrieval evaluation experiments [2]. Providing the infrastructure for large scale evaluation of retrieval methodologies, the Text REtrieval Conference (TREC) was established in 1992 to support IR researches. The Relevance judgment set is performed by human assessors appointed by TREC which called qrels. This method is costly and time consuming. There are different methods for creating relevance judgments. Researchers validate their methods for creating relevance judgment in IR

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_38, © Springer International Publishing Switzerland 2014

evaluation by measuring the agreement between the judgments generated through the proposed method and those generated via human assessors to see whether the proposed methods are reliable replacements for human assessors. Recently, the use of crowdsourcing for relevance judgment increases to conquer the problems that current evaluation methods have through expert judges.

In this work: (i) we examine the agreement between the judgments generated through crowdsourced workers and judgments generated via TREC assessors, (ii) we evaluate how the rank ordering of systems change while replacing human assessors' judgments with crowdsourcing through different evaluation metrics. The following section elaborates the related works on obtaining inter-annotator agreements and crowdsourcing approaches for creating relevance judgments. Section 3 explains the research methodology and the dataset used in the experiments. Section 4 presents and discusses the results of the experiment. Finally, the discussion and conclusion display in Section 5.

2 Related Work

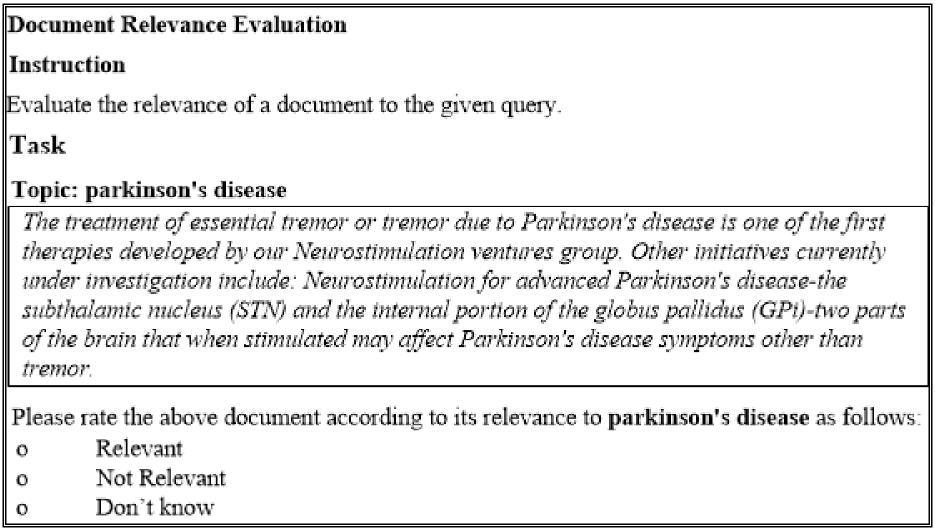
There are various studies that examine the use of crowdsourcing in information retrieval evaluation and analyze the factors that influence on the successfulness of crowdsourcing experiments. These studies can be categorized in five groups. The first group investigates the agreement between human assessors and crowdsourced workers, and examine the reliability of the judgments [3, 4]. The second group examines the effect of human factors on the accuracy of the judgments such as demographic data and personality of the workers or other human factors such as motivation and interest [5]. The third group assesses the monetary factors and its influence on the accuracy. These studies investigate whether higher payment and compensation leads to enhance the accuracy or not [6-8]. The fourth group surveys on how the design of a task and user interface may affect the results of the experiments [9-12]. Finally, the last group assesses different quality control methods to have successful crowdsourcing experiments [13-15].

The research discussed in this article is descent from the first group as it examines the reliability of the crowdsourcing for creating relevance judgments. The pioneer of using crowdsourcing for IR evaluation was Alonso and colleagues [3]. They ran five preliminary experiments by testing different alternatives such as qualification tests and changing interface through Amazon Mechanical Turk using TREC data and measured the agreement between crowdsourced workers and TREC assessors. The findings showed that judgments of crowdsourced workers were comparable with the human assessors- even if in some cases the workers detected human assessors' errors. In 2012, the use of crowdsourcing for creating relevance judgments was validated through a comprehensive experiment [4]. In order to measure the agreement, the *percentage agreement*, *fleiss's kappa* and *krippendorff's alpha* were used. The experimental results showed that crowdsourcing is a low cost, reliable and quick solution and could be considered alternative to create relevance judgment by expert assessors but it could not be a replacement for current methods due to the presence of several gaps and questions. For instance, the scalability of this approach has not been deeply investigated yet. The agreement between experts and crowdsourced workers in image annotation showed a comparable quality [1]. In extension of the work presented in [4] and [1], this study investigates the agreement between workers and TREC assessors. In this study we also examine the reliability of crowdsourcing in system ranking (how the ranking change when using crowdsourcing as a relevance judgment set). Moreover, we assess the use of crowdsourcing for creating relevance judgments for documents instead of images.

The main goal of this study is to investigate whether crowdsourcing is reliable enough to create relevance judgments for a test collection campaign. Firstly, the agreement between crowdsourced workers and TREC assessors is examined. Secondly, the rank ordering of systems is investigated when replacing human assessors' judgment set with crowdsourcing.

3 Experimental Design

The experiment was conducted in Crowdflower [16], one of the popular crowdsourcing platforms. Eight topics of TREC-9, Web Track and 20 documents were chosen for each topic selected from WT10g collection. Ten relevant and ten non-relevant documents were chosen to have a rational mix. For each of the 160 <topic, document> pairs, 9 binary judgments were collected through crowdsourcing. There are nine judgments for the same <topic, document> but from different workers. Therefore, we have 1440 judgments. Each worker should answer a relevance question and Fig. 1 shows the task as seen by the workers.



Document Relevance Evaluation

Instruction

Evaluate the relevance of a document to the given query.

Task

Topic: parkinson's disease

The treatment of essential tremor or tremor due to Parkinson's disease is one of the first therapies developed by our Neurostimulation ventures group. Other initiatives currently under investigation include: Neurostimulation for advanced Parkinson's disease-the subthalamic nucleus (STN) and the internal portion of the globus pallidus (GPi)-two parts of the brain that when stimulated may affect Parkinson's disease symptoms other than tremor.

Please rate the above document according to its relevance to **parkinson's disease** as follows:

o Relevant

o Not Relevant

o Don't know

Fig. 1. A screenshot of the task

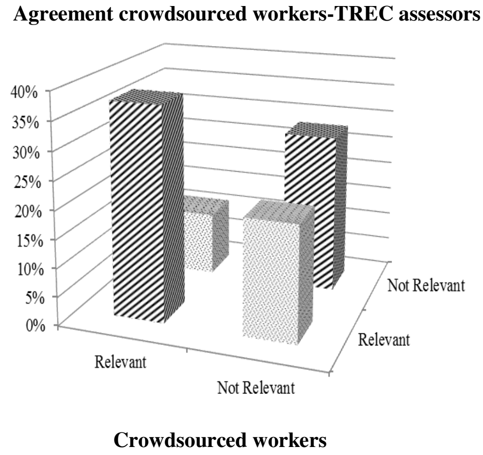
4 Results

The individual agreement between TREC assessors and workers on the binary relevance assessment is reported in section 4.1 and group agreement is discussed in section 4.2. The rank correlation is reported in section 4.3.

4.1 Individual Agreement

A first analysis concerns the individual agreement between each worker and each TREC assessor. As an instance, if the worker and TREC assessor judge the same <topic, document> similarly, they are agreed. Individual agreement means that each worker is considered individually. There are different methods to measure the agreement between each worker and TREC assessor. In this study, the individual agreement is calculated through two different methods: (i) percentage agreement and (ii) free-marginal kappa which is explained more in the following.

Percentage Agreement. This measure sums the judgments which have the same judgments by two assessors (crowdsourced workers and TREC assessors) and divide by the total number of judgments judged by two assessors. Fig. 2 displays the results and graphical representation of the percentage agreement. There is a 65.68% agreement between crowdsourced workers and TREC assessors (37.5% on relevant and 28.18% on not relevant). This results is in line with the results of [4] which found the percentage of individual agreements, 68% .



		TREC assessors (qrels)	
		R	NR
Workers	R	37.5%	20.31%
	NR	11.25%	28.18%

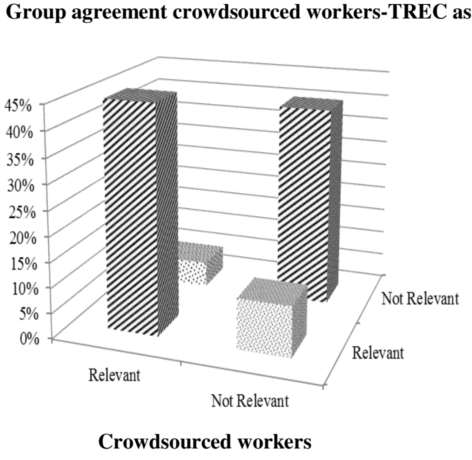
Fig. 2. Individual agreement between crowdsourced workers and TREC assessors

Free-Marginal Kappa. In order to evaluate the reliability of the agreement among assessors, kappa statistics can be used. Formerly it was proposed by Cohen [17] that utilized to compare the agreement between two assessors. In this study, Free-marginal kappa is used to measure the degree of agreement while removing the effect of random agreement. Free-marginal kappa can be used for the case of multiple judges and when the assessors are not forced to judge certain number of documents [1]. The kappa statistic is computed using an online kappa calculator [18]. If a kappa value is above 0.6, it shows an acceptable agreement. While a value above 0.8 represents perfect agreement [19]. Free-marginal kappa is 0.28 which is below the acceptance value.

In general, the individual agreement between TREC assessors and crowdsourced workers is not acceptable. Therefore, the agreement between TREC assessors and a group of workers is investigated to see whether the agreement is higher than the individual agreement.

4.2 Group Agreement

Nine workers judge the same <topic, document>, the group agreement between workers and TREC assessors is calculated to see whether the agreement is higher than individual agreement. The Majority Voting (MV) is used to aggregate the judgments. MV is a straightforward and common method which eliminates the inaccurate judgments by using the majority decision [20]. Fig. 3 shows the group agreement between TREC assessors and crowdsourced workers.



		TREC assessors (qrels)	
		R	NR
Workers	R	45%	10%
	NR	5%	40%

Fig. 3. Group agreement between qrels and workers

The group agreement between workers and TREC assessors is 85% which is higher than the individual agreement by 20%. The results of group agreement is also higher than the results of [4] which found 77% agreement between workers and TREC assessors. The kappa statistics is 0.7 that shows an acceptable agreement. In general, group agreement is more reliable than the individual agreement as we aim to replace the TREC assessors with workers.

4.3 Rank Correlation

Kendall's τ [21] is a non-parametric statistic that utilizes to assess the correlation between two ranked lists. It is one of the common statistics used in IR evaluation experiments. In this study, we use this test to compare reliability of system ranking using different relevance judgment set, TREC assessors and crowdsourced workers applying different evaluation metrics. The high correlation is considered as the same ranking in both lists. In IR evaluation, a Kendall's τ of 0.8 is considered as acceptable correlation.

In the first step, system ranking is found based on the relevance judgment set which created by TREC assessors. Then, the system ranking is created based on relevance judgment set which generated by crowdsourced workers. In the second step, the resulting ranked lists are compared to each other through Kendall τ correlation coefficient. Table 1 displays the Kendall τ correlation coefficients using two measures, Mean Recall and Mean Average Precision (MAP). The ranked list of systems is shown in Fig 4 using relevance judgment set which created by TREC assessors and crowdsourcing.

The Kendall's τ shows an acceptable correlation in ranking between crowdsourced workers and TREC assessors. So from these results, we can conclude that there is consistency between two ranked lists.

Table 1. Kendall's τ based on different metrics

	Mean Recall	MAP
Kendall's τ	0.8	0.7

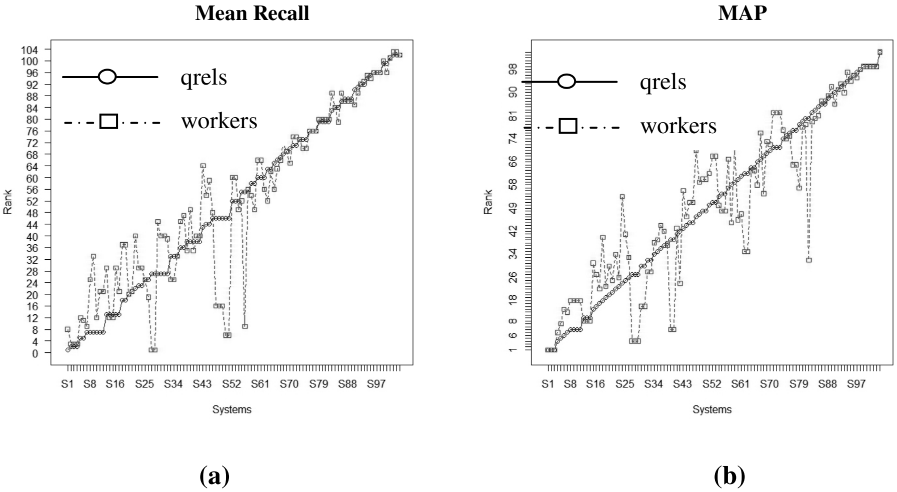
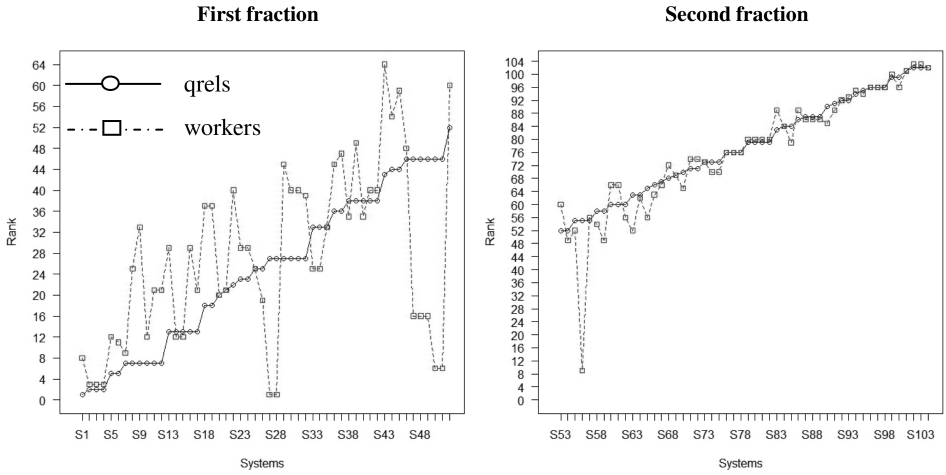


Fig. 4. a) System ranking based on TREC assessors and crowdsourced workers using Mean Recall, b) System ranking based on TREC assessors and crowdsourced workers using MAP

As Fig 4 shows, the two system ranking (TREC assessors and crowdsourced workers) are more comparable in the second fraction for Mean Recall. Therefore the tau value is calculated for two fractions of ranked list (first fraction and second fractions contains 52 systems) and show the result in Table 2. For the measure Mean Recall, the second fraction shows the higher correlation which is an acceptable correlation (see Fig 5).

Table 2. Kendall's τ value using Mean Recall for two fractions of ranked list

	First fraction (first 52 systems)	Second fraction (second 52 systems)
Kendall's τ	0.4	0.9


First fraction

Second fraction

(a) **(b)**

Fig. 5. a) System ranking using Mean Recall for the first fraction. b) System ranking using Mean Recall for the second fraction.

Using MAP in the third fraction, there is the highest correlation as you can see in the Table 3 (see Fig 6). It shows that the system ranking based on relevance judgment set generated by crowdsourced workers and TREC assessors are more consistent for low performing systems. So, relevance judgments generated through crowdsourcing produces a more reliable systems ranking when it involves measuring of low performing systems.

Table 3. Kendall's τ value using MAP for three fractions of ranked list

	First fraction (first 34 systems)	Second fraction (second 34 systems)	Third fraction (third 35 systems)
Kendall's τ	0.41	0.34	0.75

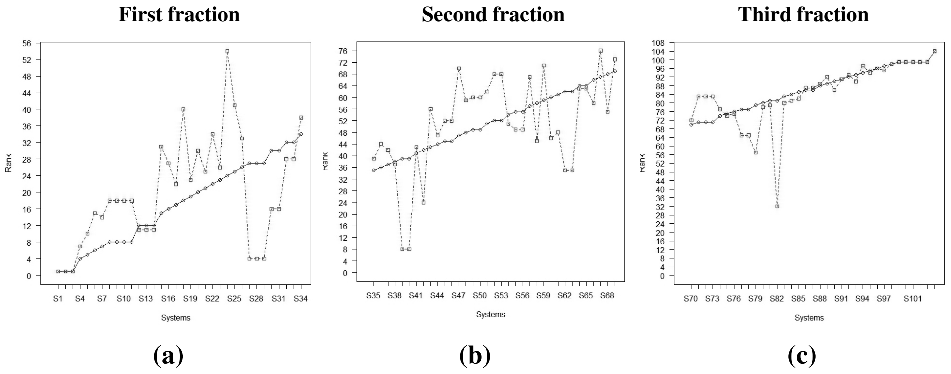


Fig. 6. a) System ranking using MAP for the first fraction, b) System ranking using MAP for the second fraction, c) System ranking using MAP for the third fraction

5 Conclusion

This study investigates the reliability of crowdsourcing for creating relevance judgment set. Different experiments on agreement between TREC assessors and crowdsourced workers are explained. The results show that when we use individual agreement, the percentage agreement between the TREC assessor and each worker is 65% and the kappa statistics show an agreement of 0.28 which is considered a low agreement, but when using group assessment, the percentage agreement between them is 85% and the kappa statistics is 0.7 which is considered as an acceptable agreement. A further experiment investigates how the rank ordering of systems change when replacing human assessors' judgment set with crowdsourcing. The results show that the relevance judgments generated through crowdsourcing produces a more reliable systems ranking when it involves measuring of low performing systems for both Mean Recall and MAP metrics. A deeper analysis about using crowdsourcing' judgment set for system ranking will be part of future work.

Acknowledgments. This research was supported by High Impact Research Grant UM.C/625/1/HIR/ MOHE/FCSIT/14 from University of Malaya and Ministry of Education, Malaysia.

References

1. Nowak, S., Rüger, S.: How reliable are annotations via crowdsourcing: a study about inter-annotator agreement for multi-label image annotation. In: Proceedings of the International Conference on Multimedia Information Retrieval, pp. 557–566. ACM (2010)
2. Cleverdon, C.: The Cranfield tests on index language devices. In: Aslib Proceedings, pp. 173–194. MCB UP Ltd. (1967)

3. Alonso, O., Mizzaro, S.: Can we get rid of TREC assessors? Using Mechanical Turk for relevance assessment. In: Proceedings of the SIGIR 2009 Workshop on the Future of IR Evaluation, pp. 15–16 (2009)
4. Alonso, O., Mizzaro, S.: Using crowdsourcing for TREC relevance assessment. Information Processing & Management 48(6), 1053–1066 (2012)
5. Kazai, G., Kamps, J., Milic-Frayling, N.: An analysis of human factors and label accuracy in crowdsourcing relevance judgments. Information Retrieval 16(2), 138–178 (2013)
6. Ipeirotis, P.: Demographics of mechanical turk (2010)
7. Ross, J., et al.: Who are the crowdworkers?: shifting demographics in mechanical turk. In: Proceedings of the 28th of the International Conference Extended Abstracts on Human Factors in Computing Systems, pp. 2863–2872. ACM (2010)
8. Kazai, G.: In search of quality in crowdsourcing for search engine evaluation. In: Clough, P., Foley, C., Gurrin, C., Jones, G.J.F., Kraaij, W., Lee, H., Mudoch, V. (eds.) ECIR 2011. LNCS, vol. 6611, pp. 165–176. Springer, Heidelberg (2011)
9. Kittur, A., Chi, E.H., Suh, B.: Crowdsourcing user studies with Mechanical Turk. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 453–456. ACM (2008)
10. Alonso, O., Baeza-Yates, R.: Design and implementation of relevance assessments using crowdsourcing. In: Clough, P., Foley, C., Gurrin, C., Jones, G.J.F., Kraaij, W., Lee, H., Mudoch, V. (eds.) ECIR 2011. LNCS, vol. 6611, pp. 153–164. Springer, Heidelberg (2011)
11. Kazai, G., et al.: Crowdsourcing for book search evaluation: impact of hit design on comparative system ranking. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 205–214. ACM (2011)
12. Alonso, O.: Implementing crowdsourcing-based relevance experimentation: an industrial perspective. Information Retrieval, 1–20 (2012)
13. Eickhoff, C., de Vries, A.P.: Increasing cheat robustness of crowdsourcing tasks. Information Retrieval, 1–17 (2012)
14. Vuurens, J.B.P., de Vries, A.P.: Obtaining High-Quality Relevance Judgments Using Crowdsourcing. IEEE Internet Computing 16(5), 20–27 (2012)
15. Allahbakhsh, M., et al.: Quality Control in Crowdsourcing Systems: Issues and Directions. IEEE Internet Computing 17(2), 76–81 (2013)
16. Crowdflower, http://crowdflower.com/
17. Cohen, J.: A coefficient of agreement for nominal scales. Educational and Psychological Measurement 20(1), 37–46 (1960)
18. Randolph, J.J.: Online Kappa Calculator (2008), http://justus.randolph.name/kappa
19. Landis, J.R., Koch, G.G.: The measurement of observer agreement for categorical data. Biometrics, 159–174 (1977)
20. Sheng, V.S., Provost, F., Ipeirotis, P.G.: Get another label? improving data quality and data mining using multiple, noisy labelers. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 614–622. ACM (2008)
21. Kendall, M.G.: A new measure of rank correlation. Biometrika 30(1/2), 81–93 (1938)

An Effective Location-Based Information Filtering System on Mobile Devices

Marzanah A. Jabar, Niloofar Yousefi, Ramin Ahmadi,
Mohammad Yaser Shafazand, and Fatimah Sidi

Faculty of Computer Science and Information Technology
Universiti Putra Malaysia 43400, Serdang, Malaysia

Abstract. As mobile devices evolve, research on providing location-based services attract researchers interest. A location-based service recommends information based on users geographical location provided by a mobile device. Mobile devices are engaged with users daily activities and lots of information and services are requested by users, so suggesting the proper information on mobile devices that reflects user preferences becomes more and more difficult. Lots of recent studies have tried to tackle this issue but most of them are not successful because of reasons such as using large datasets or making suggestions based on dynamically collected ratings within different groups instead of focusing on individuals. In this paper, we propose a location based information filtering system that exposes users preferences using Bayesian inferences. A Bayesian network is constructed with conditional probability table while Users characteristics and location data are gathered by using the mobile device. After preprocessing those data, the system integrates that information and uses time to produce the most accurate suggestions. We collected a dataset from 20 restaurants in Malaysia and we gathered behavioral data from two registered users for 7 days. We conducted experiment on the dataset to demonstrate effectiveness of the proposed system and to explain user preferences.

Keywords: Information filtering, Bayesian network, Location-based systems.

1 Introduction

Recently there has been many interest on personalized techniques due to accessibility of wireless networks and popularity of mobile devices with operating systems. The recommendation systems can be found in different applications and systems which engage users to vast collections of objects. Such systems generally provide a list of filtered information as recommended items. Those recommendations will help user to decide to choose appropriate item and help with the task of finding users preferences. For example, electronic commerce company Amazon[1] recommends similar products in order to help the user decide between

[1] http://www.amazon.com/

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_39, © Springer International Publishing Switzerland 2014

alternative products, and find preferred product. Considering recent progresses in both design and consumption of mobile devices, different resources are existing to find preferences. Such a device can be used to provide user information recommendation, using users geographical location. However, finding a proper service and information at proper time is a difficult task that requires filtering of information. Recently, there has been an upsurge of interest in the field of information filtering, mostly focusing on providing a new algorithm for recommendation systems. A system designer who seeks to add an information filtering system to her system has a vast amount of algorithm to choose, and must make a decision to find the most suitable algorithm for her needs. Generally, system designers compare those algorithm according to performance of recommenders based on results of past experiments. The designer then choose the best performing case given environmental constraints such as time, availability of data, process and memory usage, and reliability of data. Also, a system designer can compare newly chosen recommendation algorithms with previous algorithms to find the best performance.

Most recommendation algorithm have been evaluated based on their capability to predict users preferences. On the other hand, it is widely admitted that predictions are important but insufficient to set up an effective recommendation engine. In many application, especially mobile application which are main focus of this study, users benefit from information filtering systems for their daily requirements such as finding the best option for dinning. Users also are interested in saving time a critical factor. In real life environment where options change rapidly, number of new factors should be considered to find the most efficient information filtering system. Those factors are discovering options based on location of mobile device, exploring diverse options in a short time, fast response of the system to change of environment properties. Hence there is a need for an information filtering system designed for mobile devices that addresses relevant issues.

In this study, we propose an information filtering system using Bayesian networks for mobile devices. The system collects user profile and location information through the mobile device. Then it stores those parameters to train Bayesian network. When a user demands service or information, the system uses highest probability parameter provided by the Bayesian network. The results then will be presented in a map interface to provide ease of use and also to overcome limitation of display in mobile devices. To evaluate our proposed system, we examine the performance of the system on a real life dataset gathered from dining outlets in Malaysia and people who used the system to choose dining options on their mobile phones. We demonstrate effectiveness of the proposed system in terms of accuracy and retrieved options. The reminder of this paper is organized as follows. The next section is dedicated to related works. Section 3 introduces the proposed Bayesian network information filtering system and discusses the model of user preferences. In Section 4, we report the results of the experiment and evaluate the results. Finally, Section 5 concludes our study with a summary and the discussion of future work.

2 Related Works

Information flirting systems has recently attracted growing attention. While most of the previous works have mainly focused on the problem of collaborative filtering to recommend a particular product to a user [1], [2], there are several lines of related work [3], [4], [5], [6] considering two major types of collaborative filtering (i.e. Memory-based collaborative filtering and model-based collaborative filtering) which we will review in this section. A memory-based collaborative filtering system is motivated by the fact that people generally trust recommendations from one or a group of like-minded people. These type of systems try to establish a correlation between user preferences and voting patterns. A memory-based collaborative filtering system generally uses a nearest-neighbor algorithm to predict user rating based on actual ratings given by like-minded users in the system [7], [8]. NICE [9] and Wsrec [10] are two examples of memory-based collaborative filtering system. Such correlation should be automatically calculated within groups of users and individuals. This makes huge load on the system, especially when results is supposed to be carried in mobile devices and real-time [11].

In contrast, model-based collaborative filtering focuses on descriptive model of an individuals preferences and then uses a machine learning algorithm for predicting voting. Examples of machine learning algorithms include linear classifier [12], dependency networks [13], Bayesian network [14]. The authors of [15] compare and contrast performance and effectiveness of mentioned learning algorithm for model-based collaborative filtering. They explore and classify the issues raised by using those algorithm on memory-based collaborative filtering. Based on the results of several experiments, they conclude that linear classifier and dependency networks are not as effective for model-based collaborative filtering as they have been for other methods. Based on the results, we preferred to use Bayesian networks in model-based collaborative filtering which is also more appropriate for mobile devices which requires real time computation.

In mobile devices, the context can information in shape of time, gathered from operating system of the mobile device, and location, gathered from Global Positioning system (GPS). Context-aware computing is a part of mobile devices that is defined specifically to location awareness [16]. Tour recommendation systems [17], commercial recommendations [18], and restaurant recommendations [19] are services that use context aware computing. Two property of context aware computing, i.e. time and location, can be recorded using information of the mobile device. However, there are challenges such as limitation of computation and memory capacity or latency [20] that makes direct recommendation on mobile devices a difficult task. Table 1 summarizes some of the information filtering systems for mobile devices.

3 Location-Based Information Filtering System

As shown in Fig. 1, the proposed location-based information filtering system contains three phases: user profile and context information collection, information

Table 1. Information Filtering Systems

	YouTube [21]	PILGRIM [22]	Fuzzy-genetic [23]	WebPUM [24]
Application type	Video	Automated system	Automated system	Automated system
Filtering type	Memory-based	Model-based	Memory-based	Model-based
Parameters	click through rate, time, user profile	User location	Demographical data, rating, online behavior	User log files
Device	Web	Mobile	Web	Web

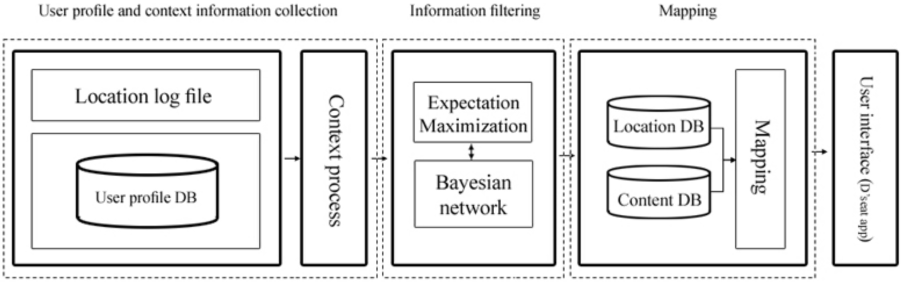


Fig. 1. Overview of the information filtering system

filtering, and mapping. User profile information are collected by using registration information provided by the user. The Bayesian network uses this information to make conditional probability table. Upon receiving a new service request by the user, the system selects highest probability node of each attribute using a dataset of context information.

3.1 Phase 1, User Profile and Context Information Collection

This phase contains the task of gathering user profile and context information from the mobile device. User profile information include age, gender and context information include user location, time, weather, season which are gathered from the mobile device. Upon a service request, the system integrates profile and context information to process information filtering.

3.2 Phase 2, Information Filtering

Once the information is gathered, we use them as input for Bayesian network. The Bayesian network has been constructed by an expert, as illustrated in Fig. 2. Since we assume user location can change, we used a maximum likelihood learning technique known as Expectation Maximization (Algorithm 1.). The expectation maximization algorithm is generally used to find the maximum likelihood parameters, i.e. location, of a statistical model in cases where the model does not reflect environmental changes.

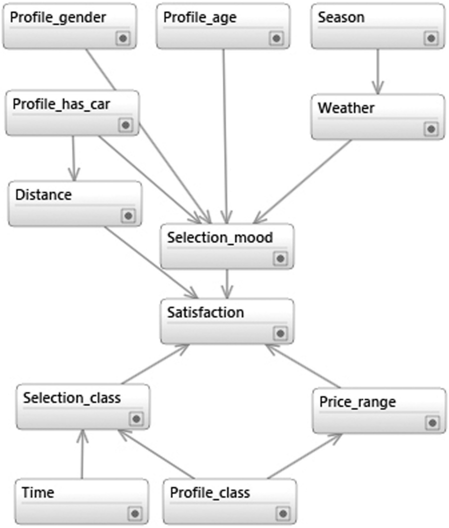


Fig. 2. Structure of Bayesian network

3.3 Phase 3, Mapping

After acquiring recommendations from The Bayesian network, the system displays information to guide the user through alternatives. The information filtering system has been implemented on a mobile application called Dseat application. The Dseat application is a free restaurant application which provides dining options for users. The mobile application uses Google Map API[2] , which is a publicly free accessible service provided by Google. In order to model personal preferences, the application collects users demographic information. The application also contains a database to store those information. Results of the Bayesian inference are used to map user preferences to the alternatives which in this case are preferred dining options. In order to do that, we construct a restaurant class which contains three parameters type, price, and mood which are suggested by authors of [21]. A user willingness to eat in a restaurant is calculated as follows:

$$X_{ijk} = (t_i \times w_t) + (p_i \times w_p) + (m_i \times w_m) \ . \tag{1}$$

Where X_{ijk} is preference parameter of a restaurant, $type = (t_1, t_2, t_3, \ldots, t_n)$, $price = (p_1, p_2, p_3, \ldots, p_n)$, $mood = (m_1, m_2, m_3, \ldots, m_n)$ and are respectively the restaurant type (e.g. Chinese, French), price of the food (i.e. Low, mid, high), mood (e.g. Romantic, fast food, normal, tidy). $weight = (w_t, w_p, w_f)$ is

[2] https://developers.google.com/maps/

the weight of the Bayesian networks node. Finally, recommended restaurant will be suggested as follows:

$$recommendation = max(X_{ijk}) \ . \tag{2}$$

Equation 1 specifies that maximum value for preference parameter is recommended to user by depicting its location on the map.

Algorithm 1. Expectation maximization algorithm

Input:
k: Number of clusters
$y = (y_1, y_2, y_3, \ldots, y_n)$ Set of -dimensional points
a : Likelihood
max : Limitation
Output:
c, r, w: Updated parameters
x: Probability matrix

```
1.initialization: set values
2.While: $\delta$(g)>$a$ & !$max$
3.Do
```

```
$k' = l' = w' = llh = 0;$
for i=1 to n
    $\sum p_i$=0;
    for $j$=1 to $c$
        $\delta_{ij} = (s_i - c_j)^t R^{-1} (s_i - c_j)$
        $p_{ij} = \frac{w_j}{2\times 3.14^{\frac{p}{2}}} exp(-\frac{1}{2}\delta_{ij})$
        $\sum p_i = \sum p_i + p_{ij}$
    end-for
    $x_i = \frac{p_i}{\sum p_i}$
    $g = g + ln(\sum p_i)$
    $k' = k + s_i x_i^t$
    $w' = w + x_i$
end-for
for $j$=1 to n
    $k_i = \frac{k_i}{w_i}$
    for $j$=1 to n
        $l' = l' + (s_i - k_i) x_{ij} (s_i - k_j)^t$
    end-for
end-for
$l = \frac{l'}{n}$
$w = \frac{w'}{n}$
```

Table 2. User Demographic Information

	Subject 1	Subject 2
Gender	Male	Female
Age	26	25
Possession of a vehicle	Yes	Yes
Type of food	American, French, Chinese, Korean, Japanese, Malaysian, Middle eastern	Chinese, Japanese, fast food
Income (RM)	4000-4500	2500

Table 3. Information Filtering Systems

	Parameter	Value
Time	Season	Spr, Sum, Aut, Win
	Period	Morning, noon, night
Preferences	Type	American, French, Chinese, Japanese, Malaysian, Korean, Middle eastern
	Price	Low, mid, high
	Mood	Romantic, fast food, normal, tidy
Demographic	Gender	Male, Female
	Age	15-99
	Possession of a vehicle	Yes, No
	Type of food	American, French, Chinese, Korean, Japanese, Malaysian, Middle eastern
	Income	1000-99000
Location	Distance	Close (100M), Mid (200M), Far (300M)
Weather	Type	Sunny, rainy, cloudy, snow

4 Experimental Results

The main task which effect the information filtering system′s effectiveness is delivering user preferences in real-time. We briefly the dataset we used in our experiment and discuss evaluation of this task. We used a dataset of 20 restaurant in Malaysia within a week (1/10/2013, 8/10/2013). We evaluated our system by conducting the learning algorithm in cross-validation model on two individual subjects. User demographic information requested by the application include gender, age, possession of a vehicle, monthly income, and type of food she prefers, her income as presented in Table 2. Factors for food preferences have been studied by [25]. We added possession of a vehicle because the system is able to locate the users location on the map. As presented in Table 3, processed data set includes time, user preferences (as mentioned in Eq. 1), weather, and finally location. Time parameter consists of time of the year (season) and time of the day to eat (period). User requests are obtained using the Dseat application. Location parameter has three attributes Close, mid, and far which are respectively 100, 200, 300 meters from current location of the existing user. Final parameter is weather which is obtained from a weather web service from open weather map [26]. This service can provide weather information via a JSON string. According to open weather map API, weather is categorized to sunny, rainy, cloudy, and snow. Parameters were inferred by the Bayesian network. Probability distribution of each parameter is measured. Accordingly, highest values of those parameters were selected by the system to be presented as recommended place. For example, subject 1 prefers French, low price and fast food for breakfast. Japanese restaurant, mid-price and normal mood for lunch and for dinner, Korean, mid-price and tidy, as shown in Table 3.

5 Conclusion and Future Work

A In this paper we addressed the problem of information filtering for mobile devices. Our proposed model-based collaborative filtering system contains three phases: user profile and context information collection, information filtering, and mapping. The system uses Bayesian network to inference user preferences. The unique contribution of this study is that it uses a maximum likelihood learning technique to overcome issue of change of location. The system was implemented in a dining phone application to find nearest restaurants which match with users food preferences. Parameters like type of restaurant, time, weather, location and mood were engaged in the system to recommend best dining options to the user. Our experimental evaluation on a real-life dataset gathered from 20 restaurants in Malaysia shows the effectiveness of our proposed system. None of the previous works consider mobility as a parameter in their information filtering systems, in this paper we proposed to use mobility as a parameter to find user preferences. However, we consider more complicated learning technique as a potential future work.

References

1. Mehta, B., Hofmann, T., Nejdl, W.: Robust collaborative filtering. In: Proceedings of the 2007 ACM Conference on Recommender Systems, pp. 49–56. ACM, Minneapolis (2007)
2. Kim, H.-N., Ji, A.-T., Ha, I., Jo, G.-S.: Collaborative filtering based on collaborative tagging for enhancing the quality of recommendation. Electronic Commerce Research and Applications 9, 73–83 (2010)
3. Bobadilla, J., Ortega, F., Hernando, A., Arroyo, Ã.: A balanced memory-based collaborative filtering similarity measure. International Journal of Intelligent Systems 27, 939–946 (2012)
4. Zhi-Dan, Z., Ming-Sheng, S.: User-Based Collaborative-Filtering Recommendation Algorithms on Hadoop. In: Third International Conference on Knowledge Discovery and Data Mining, WKDD 2010, pp. 478–481 (2010)

5. SongJie, G., HongWu, Y., Hengsong, T.: Combining Memory-Based and Model-Based Collaborative Filtering in Recommender System. In: Pacific-Asia Conference on Circuits, Communications and Systems, PACCS 2009, pp. 690–693 (2009)
6. Wi, H., Oh, S., Mun, J., Jung, M.: A team formation model based on knowledge and collaboration. Expert Systems with Applications 36, 9121–9134 (2009)
7. BellogÃ-n, A., Wang, J., Castells, P.: Bridging memory-based collaborative filtering and text retrieval. Inf. Retrieval, 1–28 (2012)
8. Koren, Y., Bell, R.: Advances in Collaborative Filtering. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 145–186. Springer US (2011)
9. Halfaker, A., Song, B., Stuart, D.A., Kittur, A., Riedl, J.: NICE: social translucence through UI intervention. In: Proceedings of the 7th International Symposium on Wikis and Open Collaboration, pp. 101–104. ACM, Mountain View (2011)
10. Zibin, Z., Hao, M., Lyu, M.R., King, I.: WSRec: A Collaborative Filtering Based Web Service Recommender System. In: IEEE International Conference on Web Services, ICWS 2009, pp. 437–444 (2009)
11. Baltrunas, L., Ludwig, B., Peer, S., Ricci, F.: Context relevance assessment and exploitation in mobile recommender systems. Personal Ubiquitous Comput. 16, 507–526 (2012)
12. Zhang, T., Iyengar, V.S.: Recommender systems using linear classifiers. J. Mach. Learn. Res. 2, 313–334 (2002)
13. Chao, C., Helal, S., de Deugd, S., Smith, A., Chang, C.K.: Toward a collaboration model for smart spaces. In: 2012 Third International Workshop on Software Engineering for Sensor Network Applications (SESENA), pp. 37–42 (2012)
14. de Campos, L.M., FernÃ!'ndez-Luna, J.M., Huete, J.F., Rueda-Morales, M.A.: Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks. International Journal of Approximate Reasoning 51, 785–799 (2010)
15. Cacheda, F., Carneiro, C., Fernandez, D., Formoso, V.: Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems. ACM Trans. Web 5, 1–33 (2011)
16. Adomavicius, G., Tuzhilin, A.: Context-Aware Recommender Systems. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 217–253. Springer US (2011)
17. Ge, Y., Liu, Q., Xiong, H., Tuzhilin, A., Chen, J.: Cost-aware travel tour recommendation. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 983–991. ACM, San Diego (2011)
18. Duan, L., Street, W.N., Xu, E.: Healthcare information systems: data mining methods in the creation of a clinical recommender system. Enterprise Information Systems 5, 169–181 (2011)
19. Park, M.-H., Park, H.-S., Cho, S.-B.: Restaurant Recommendation for Group of People in Mobile Environments Using Probabilistic Multi-criteria Decision Making. In: Lee, S., Choo, H., Ha, S., Shin, I.C. (eds.) APCHI 2008. LNCS, vol. 5068, pp. 114–122. Springer, Heidelberg (2008)
20. Yang, F., Wang, Z.: A mobile location-based information recommendation system based on GPS and WEB2.0 services. Database 7, 8 (2009)
21. Davidson, J., Liebald, B., Liu, J., Nandy, P., Vleet, T.V., Gargi, U., Gupta, S., He, Y., Lambert, M., Livingston, B., Sampath, D.: The YouTube video recommendation system. In: Proceedings of the Fourth ACM Conference on Recommender Systems, pp. 293–296. ACM, Barcelona (2010)

22. Brunato, M., Battiti, R.: PILGRIM: A location broker and mobility-aware recommendation system. In: Proceedings of the First IEEE International Conference on Pervasive Computing and Communications (PerCom 2003), pp. 265–272 (2003)
23. Al-Shamri, M.Y.H., Bharadwaj, K.K.: Fuzzy-genetic approach to recommender systems based on a novel hybrid user model. Expert Systems with Applications 35, 1386–1399 (2008)
24. Jalali, M., Mustapha, N., Sulaiman, M.N., Mamat, A.: WebPUM: A Web-based recommendation system to predict user future movements. Expert Systems with Applications 37, 6201–6212 (2010)
25. Harrington, R.J., Ottenbacher, M.C., Kendall, K.W.: Fine-Dining Restaurant Selection: Direct and Moderating Effects of Customer Attributes. Journal of Foodservice Business Research 14, 272–289 (2011)
26. Nyrhinen, F., Salminen, A., Mikkonen, T., Taivalsaari, A.: Lively Mashups for Mobile Devices. In: Phan, T., Montanari, R., Zerfos, P. (eds.) MobiCASE 2009. LNICST, vol. 35, pp. 123–141. Springer, Heidelberg (2010)

An Enhanced Parameter-Free Subsequence Time Series Clustering for High-Variability-Width Data

Navin Madicar, Haemwaan Sivaraks, Sura Rodpongpun,
and Chotirat Ann Ratanamahatana

Dept. of Computer Engineering, Chulalongkorn University
254 Phayathai Rd., Pathumwan, Bangkok, Thailand, 10330
{navin.ma,haemwaan.s}@student.chula.ac.th,
{g53srd,ann}@cp.eng.chula.ac.th

Abstract. In time series mining, subsequence time series (STS) clustering has been widely used as a subroutine in various mining tasks, e.g., anomaly detection, classification, or rule discovery. STS clustering's main objective is to cluster similar underlying subsequences together. Other than the known problem of meaninglessness in the STS clustering results, another challenge is on clustering where the subsequence patterns have variable lengths. General approaches provide a solution only to the problems where the range of width variability is small and under some predefined parameters, which turns out to be impractical for real-world data. Thus, we propose a new algorithm that can handle much larger variability in the pattern widths, while providing the parameter-free characteristic, so that the users would no longer suffer from the difficult task of parameter selection. The Minimum Description Length (MDL) principle and motif discovery technique are adopted to be used in determining the proper widths of the subsequences. The experimental results confirm that our proposed algorithm can effectively handle very large width variability of the time series subsequence patterns by outperforming all other recent STS clustering algorithms.

Keywords: Subsequence Time Series (STS) Clustering, Time Series, Variable-width, Parameter-free, MDL.

1 Introduction

Time series data is used widely in many fields such as finance, image recognition, medicine, etc. Accordingly, Time series data mining [5] becomes prevalent in the field of knowledge discovery, as can be seen through various mining tasks, i.e. clustering, classification [10], rule discovery [2], motif discovery, among many others. The main aspect of this paper is on time series clustering, and we pay special attention to one of the clustering techniques called subsequence time series (STS) clustering.

In any single time series sequence, its subset components are called subsequences. The main purpose of STS clustering is to cluster these subsequences together by looking at the similarity among subsequence pairs, as shown in Fig. 1.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_40, © Springer International Publishing Switzerland 2014

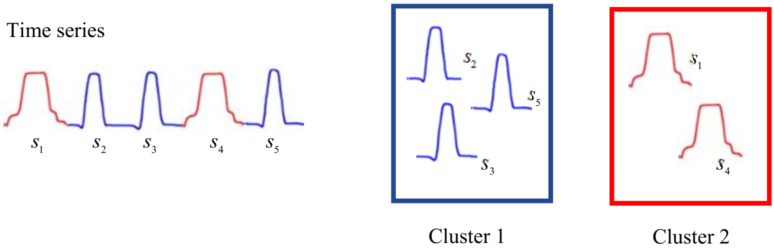


Fig. 1. Example of STS clustering. Given the time series sequences s_1, s_2, s_3, s_4 and s_5. After clustering, s_2, s_3 and s_5 are grouped together in cluster 1 while s_1 and s_4 are in cluster 2.

In a first glance, this may seem quite straightforward, especially if we know the size of the subsequence patterns. However, this is often not the case in practice when the time series sequence may consist of several patterns of various sizes or the exact size of the patterns is not known. Almost all of the previous work on STS clustering are based on a fixed width of the pattern. There have been just a few that tried to resolve this problem [6][8]. However, none of them are flexible enough to handle general variable-width subsequence patterns, as they only work under some predefined parameters. In this work, we attempt to eliminate these hard-to-set parameters, while maintaining highly accurate STS clustering results. This parameter-free characteristic takes away all the hassles for users in figuring out the proper parameter settings for the algorithm. Our preliminary work [12] has shown promising results that parameter-free STS clustering is in fact feasible.

Therefore, this work attempts to perfect up the approach by proposing an improved parameter-free framework that can maintain high clustering accuracy with some width variability in the subsequence patterns using hierarchical clustering adopted from [14]. We first build an initial cluster through motif discovery, a task of finding two most similar subsequences in time series, and then choose an option that yields minimum cost among building a new cluster, adding a subsequence into an existing cluster, and merging two existing clusters. Repeat this step until the stopping criterion is reached. Additionally, Minimum Description Length (MDL) principle [4] is utilized and incorporated into the motif discovery process to determine that proper widths of the patterns. As will be shown in the experiment results, comparing with the previous works, our novel approach gives more accurate results, while maintaining the parameter-freeness property.

The rest of the paper is organized as follows. Section 2 gives some reviews on previous work in STS clustering, as well as providing some background knowledge of the problem and the definitions of terminologies used in this work, Section 3 describes our proposed algorithm. Section 4 shows the experimental results to demonstrate the effectiveness of our algorithm, and finally, Section 5 provides the conclusion and discussion.

2 Related Work, Definition and Background

2.1 Related Work

The two most recent works on STS clustering [6][8] have shown to give good clustering results. Both of the approaches are based on the following similar ideas. First, if every single subsequence in a time series is used, clustering results will be meaningless because many non-meaningful or noisy parts are included. Therefore, ignoring some subsequences is needed. Second, the subsequences to be clustered are not allowed to overlap each other to avoid meaningless clustering result that was discovered in [2] by [1]. However, even though both of the works can resolve the meaninglessness in STS clustering problem, some predefined parameters are still needed to be determined by domain experts or users. In particular, the work by [8] needs two parameters '*w*' and '*f*' to define the range of the subsequences' width to be *w*/*f* to *w***f* while [6] needs the parameter '*s*' as the approximate width whose range is from *s* to 2*s*. Apart from these hard-to-set parameters by users, the variability of the subsequence' widths is limited. Therefore, in this work, we are proposing a parameter-free algorithm that could fully support variability in the subsequences' widths.

In our previous work [12], to find out what the proper widths are, we run motif discovery algorithm at all possible widths (2 data points to half of the time series' length), and then use some statistical principles to determine and rank the motifs by priority based on similarity and frequency of those motifs occurring at the same location. After that, we get some promising widths, and we use them to cluster subsequences in clustering process. Unfortunately, we found that this approach loses some potential patterns, and it slightly lowers the accuracy in some cases, as will be illustrated in Section 4. In this work, we therefore propose a new solution that eliminates the problem and also improves the variability.

2.2 Definition

Definition 1: A *time series T* of length *n* is a set of *n* real values in a sequence, defined as $T = \{t_1, t_2, \ldots, t_n\}$.

Definition 2: A *subsequences S* of length *m* of the time series *T* of length *n* is any *m*-length subsets of *T*, defined as $S_i^m = \{t_i, t_{i+1}, \ldots, t_{i+m-1}\}$, where $1 \le i \le n\text{-}m\text{+}1$ and $m < n$.

Definition 3: A *motif M* of length *w* is a set of the two most similar subsequences of length *w* of any time series, defined as $M_w = \{S_i^w, S_j^w\}$ where $i < j$ and $i\text{+}w\text{-}1 < j$. *The similarity measure is calculated by the well-known Euclidean distance.

Definition 4: A *Cluster C* is a set of subsequences which are assigned to the same group, defined as $C = \{S_1, S_2, \ldots, S_n\}$ where *n* is the number of subsequences in *C*.

Definition 5: A *Cluster Center* of cluster *C* is the representative subsequence of *C* determined from the average values of all subsequences in *C*, defined as $C_{center} = \{s_1, s_2, \ldots, s_m\}$ where $s_i = (S_{1_i} + S_{2_i} + \ldots + S_{n_i}) / n$ and $1 \le i \le m$.

2.3 Background

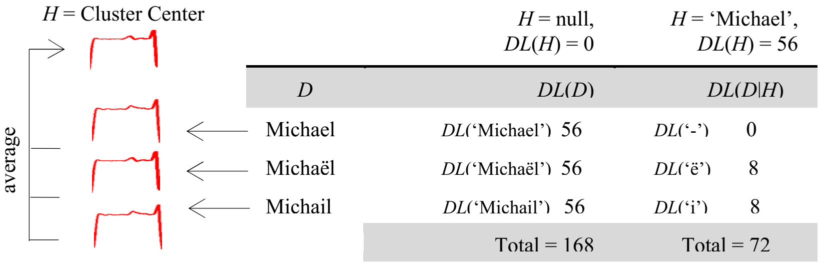
The background knowledge that is required to understand the algorithm includes motif discovery technique and MDL principle. These are the essences of our proposed work that would help determine the proper widths of the subsequences.

Motif discovery is one of the time series mining tasks typically used for finding the two most similar subsequences in a time series. The well-known motif discovery is Mueen-Keogh (MK) motif discovery algorithm [7] which is claimed to be the fastest exact algorithm available to date. However, this algorithm still needs a parameter '*w*' to define the width of the subsequence patterns, and then all possible subsequence patterns of that width are determined for finding the most similar pair. This technique essentially reduces the searching time from the brute-force approach down to a satisfactory level.

MDL principle is an important concept in the field of information theory that helps us find the hypothesis with the best ability to compress on a given dataset. How could it help in time series sequences? To help develop an intuition, we first show an example in discrete data [6].

> Ex. Given the name 'Michael' in various languages: Michael (English), Michaël (Dutch) and Michail (Greek). If 8 bits are needed for each character to store the data, so-called the description length of the data (*DL*(*D*) where *D* is the data), normally we use 8*7=56 bits for each name and 56*3=168 bits for all of them. But with MDL, we are determining the hypothesis *H* of having 'Michael' as the name. We do not have to store all of them but only the hypothesis (*DL*(*H*)) and the difference between the hypothesis and all elements in the class (where *A* is an element in class *C*) are needed, as shown in Fig. 3.
>
> $$\sum_{A\in C} DL(A\,|\,H)$$

Through this technique, we use only 72 bits that makes the 168-72=96 bitsave (*DL*(*Before*)-*DL*(*After*)). Notice that the bitsave indicates the compression ability of the hypothesis. Now, we if we could map the discrete data to the time series data, the same idea still applies. Suppose that the hypothesis is a cluster center. It let us know that the more compression ability that cluster center has, the more similarity of the subsequences would be. This is the main reason why MDL suits our purpose.



	H = null, $DL(H) = 0$	H = 'Michael', $DL(H) = 56$
D	$DL(D)$	$DL(D\|H)$
Michael	DL('Michael') 56	DL('-') 0
Michaël	DL('Michaël') 56	DL('ë') 8
Michail	DL('Michail') 56	DL('i') 8
	Total = 168	Total = 72

Fig. 2. Example of MDL principle of how to adapt the idea to time series data. Bitsave value is used to measure the similarity of elements in the class.

3 Algorithm

Our algorithm consists of two phases. The first phase is the selection of the proper subsequences' widths where both motif discovery technique and MDL principle are applied. We get the set of motifs with their potential widths. And in the latter phase, some of them are used as the initial clusters to guide the direction of the clustering process. The detail of each phase is described as follows.

3.1 First Phase: Selection of the Proper Widths

The idea of using motif discovery technique together with MDL principle is reproduced from [11] to find the set of proper motifs with their potential widths without requiring any predefined parameters. Similar to our previous work, MK motif discovery is run at all possible widths but MDL is used instead to rank priority of the motif results. The higher the compression ability, the higher quality those motifs are. However, simply applying these ideas are not enough to acquire accurate results we expect because there are some special cases that further handling must be applied. Fig. 3 illustrates an example of scenario where inaccurate motifs are discovered and could mistakenly be added into the group.

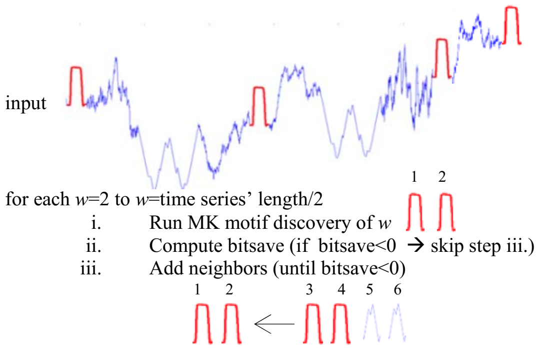

input
1 2
for each w=2 to w=time series' length/2
i. Run MK motif discovery of w
ii. Compute bitsave (if bitsave<0 → skip step iii.)
iii. Add neighbors (until bitsave<0)
1 2 3 4 5 6

Fig. 3. Example of the first phase's main process. Given the input time series, MK motif discovery is run in all widths, and for each *w* bitsave and neighbors of the motif are considered.

After the motif is found in each width, bitsave is calculated as follows.

$$Bitsave = \sum_{S \in M} DL(S) \; - \; (DL(H) + \sum_{S \in M} DL(S \mid H)) \qquad (1)$$

where *S* are two subsequences of motif *M*, *H* is the average subsequence of those two subsequences called motif center (similar to the cluster center in Definition 5), and *DL* of any time series or subsequences is defined as follows.

$$DL(T) = m * - \sum_{t} P(T = t) \log_2 P(T = t) \qquad (2)$$

where T is a time series or subsequences, t is a unique value in time series T, $P(T = t)$ is the probability that t occurs in T, and m is the length of time series T. Note that in general, the time series is discretized into a 64-bit representation before the calculation. See the details in [6].

If the bitsave is not larger than zero, that means the subsequences of the motif are not considered similar, and we would discard that motif because it is no longer interesting. Next step is finding their other similar subsequences, called neighbors, through bitsave calculation again. But this time, the following Bitsave formula is instead used.

$$Bitsave = DL(S') - DL(S' \mid H) \tag{3}$$

where S' is new subsequences to be added ((1) and (3) are from the original formula which is *Bitsave*=*DL*(*Before*)-*DL*(*After*))

We collect the neighbors until there is no more bitsave attained (bitsave ≤ 0). However, in Fig. 3, step iii., we can notice that the subsequences 5 and 6 should not be the neighbors because of their dissimilarity from others, as re-illustrated in Fig. 4. This is crucial in improving the clustering results.

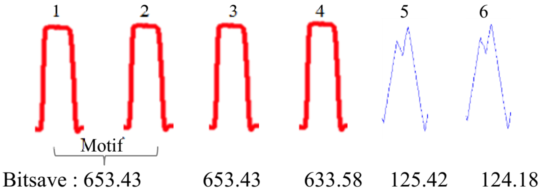


Fig. 4. Problem of the dissimilarity among neighbors. According to the original algorithm, subsequences 5 and 6 are considered neighbors (because bitsave > 0), but we can visually see that they should be considered different.

We can see from the figure that their bitsaves are noticeably different. To resolve this, we adopt a technique from [3]. We plot all bitsave values in the straight line, and we can figure out the point where it can best split all of the values into two groups by maximizing the following gap.

$$\text{Gap} = \mu_R - \sigma_R - (\mu_L + \sigma_L) \tag{4}$$

where μ_R and σ_R are mean and standard deviation of bitsave values in the right group, and μ_L and σ_L are those in the left group.

After the split is acquired, the subsequences are separated , and we keep only the right-side subsequences with higher bitsave values, as shown in Fig. 5.

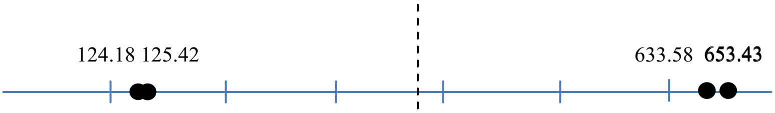


Fig. 5. Bitsave in the straight line are split into two sides. Only the right side values are kept, and the left side values are discarded.

Now, we obtain a higher-quality set of motifs but their number is still very large, but only the interesting motifs are needed. Note that the clustering accuracy greatly depends on the result from this phase, so we have to be particularly selective to achieve good results. As we want the motifs with high compression ability, and it is much better if they also have an exclusive shape, we can utilize the resulting neighbors from the previous step. We start with sorting them by their bitsave values, and then discarding the motifs in lower ranks that are subset or superset of the higher-rank motifs and their neighbors since they contribute to the same shape.

3.2 Second Phase: Clustering Process

In clustering process, we follow our previously proposed steps and algorithms [12]. This clustering process requires the results from the previous phase to be used as initial clusters. Hierarchical clustering is used, as shown in Fig. 6, where each internal node needs to decide the operation for the subsequences: ‘Create’ is divided into 2 types. Type I is creating an initial cluster from the best motif of the results from the first phase. Type II is creating another cluster with the same width of an existing initial cluster. ‘Add’ is adding the most similar subsequences into an existing cluster. ‘Merge’ is combining two existing clusters together. In the first round, Type I Create is required since an initial cluster must be created. In other successive rounds, the choice with minimum error is selected until the stopping criterion is reached.

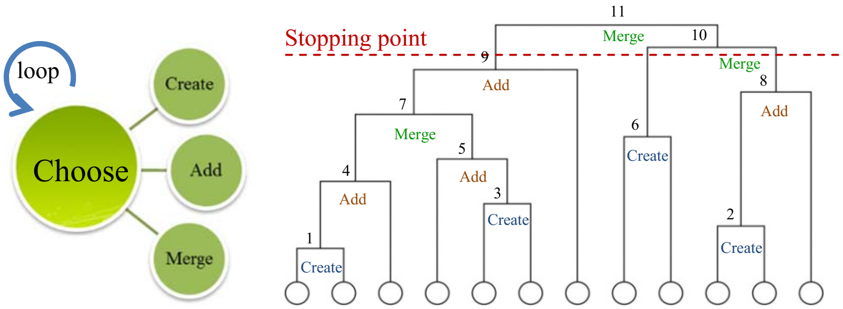


Fig. 6. Model of hierarchical clustering. The elements in the lowest level represent the subsequences, and each step labeled by the number is the operation chosen.

4 Experimental Results

Datasets from the UCR time series clustering/classification archive [9], the largest time series data archive, are used to test both validity and accuracy of our algorithm. Because we know the expected patterns exactly, Accuracy-on-Detection (AoD) [13] is employed as a measurement of percentage of correction and overlap between output subsequences and expected (or ground truth) subsequences, defined as follows.

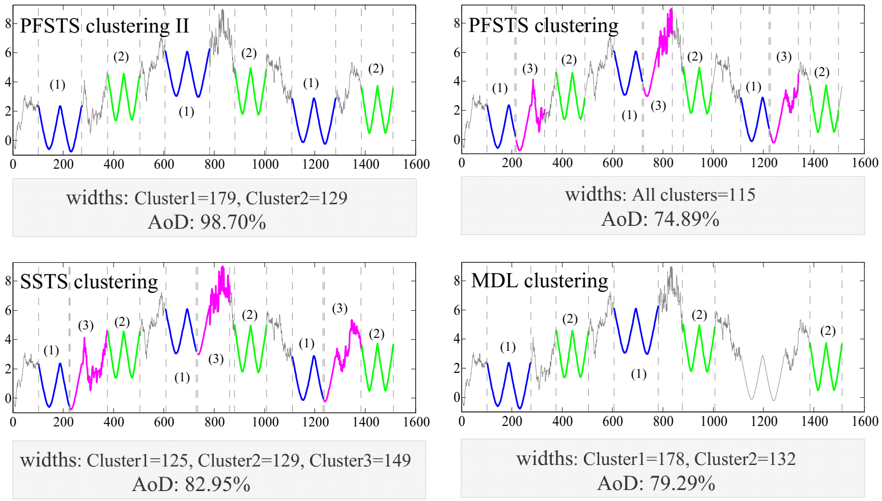
$$AoD = \frac{\sum_{j=1}^{k_R} \sum_{s \in C_j, r \in R_j} O(s,r)}{\sum_{j=1}^{k_R} \sum_{s \in C_j, r \in R_j} U(s,r)} \text{ x } 100\% \tag{5}$$

where $C = \{C_i \mid 0 < i \leq k_C\}$; k_C is the number of clusters, the expected result is $R = \{R_j \mid 0 < j \leq k_R\}$; k_R is the number of clusters, C_i and R_j are the clusters (in Definition 4) containing some subsequences as its cluster members. $O(s,r)$ and $U(s,r)$ are overlap and union between $s \in C_i$ and $r \in R_j$ respectively, and at each s which is placed in a wrong cluster, $O(s,r)$ is defined as 0 and $U(s,r)$ is $U(r,r)$.

Due to the variety of data patterns, the results are shown in two cases by the variability of data. First is the case where variability of the widths in different subsequence patterns is small. In this case, we compare our result (PFSTS clustering II) to the following previous algorithms we mentioned earlier, SSTS clustering [8], MDL clustering [6], and the original PFSTS clustering [12].. Second is the case where variability of the widths in different subsequence patterns is quite large. In this case, SSTS [8] and MDL [6] clusterings failed to give the results due to their exceedingly large time and space complexity. Therefore, only the PFSTS clustering lasts to compare.

4.1 Time Series with Small Variability in the Subsequence Widths

The dataset we used consists of two classes of subsequences, one class of width 128 data points, and the other of width 176 data points. Between subsequence patterns, the random walk data of length 100 data points are placed. In SSTS clustering and



widths: Cluster1=179, Cluster2=129
AoD: 98.70%

widths: All clusters=115
AoD: 74.89%

widths: Cluster1=125, Cluster2=129, Cluster3=149
AoD: 82.95%

widths: Cluster1=178, Cluster2=132
AoD: 79.29%

Fig. 7. Comparing results from four STS clustering algorithms. It shows that our algorithm, PFSTS clustering II, gets the best result.

MDL clustering that need the predefined parameters, we set them such a way to guarantee that their algorithms will discover the patterns successfully to give the best benefits to both rival methods (in this dataset, the widths range from 128 to 178 data points). The results are shown in Fig. 7 with discovered widths and AoD.

Note again that both PFSTS clustering II and PFSTS clustering are parameter-free in nature, so the widths shown are the proper widths that the algorithms have automatically chosen. Also, MDL and SSTS both have to choose the proper widths within small range over the actual width, and we provided them the parameters that are most advantageous to them. Nevertheless, our proposed algorithm outperforms all of them, giving the most reasonable widths and highest AoD.

4.2 Time Series with High Variability in the Subsequence Widths

In this case, two datasets are provided to strongly support our variable-width functionality, which is the main contribution of the work. Unfortunately, as mentioned earlier, we can only compare our results to those from PFSTS clustering because both MDL and SSTS algorithms require exceedingly high time and space complexity that no results could be produced within reasonable amount of time. The results are shown in Fig. 8. and Fig. 9. where the data used in Fig. 8. consists of two classes of subsequences of lengths 286 and 576 data points with the random walk data of length 250 data points placed between them, and the data used in Fig. 9. consists of three classes of subsequences of lengths 128, 275 and 1050 data points with the random walk data of length 200 data points.

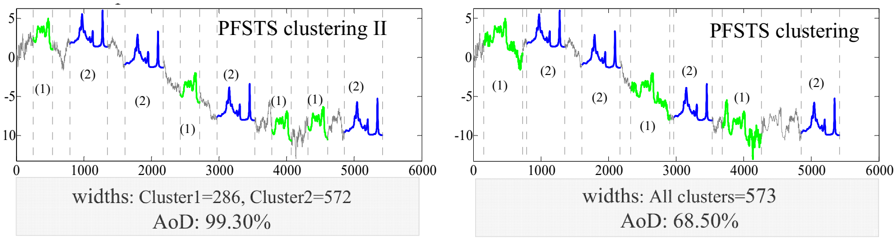


Fig. 8. Comparing results between PFSTS clustering II and PFSTS clustering. The dataset consists of two classes with high variability in the subsequence widths.

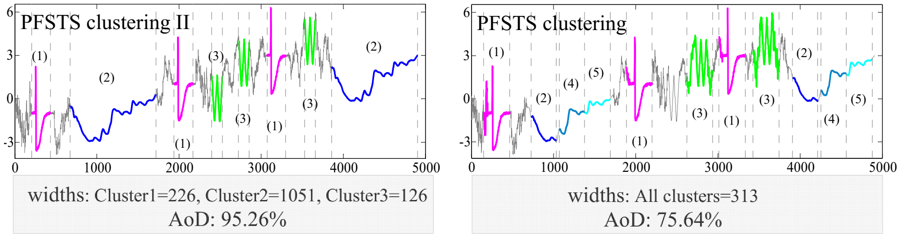


Fig. 9. Comparing results between PFSTS clustering II and PFSTS clustering. The dataset consists of three classes with high variability in the subsequence widths.

Figs. 8 and 9 show that the results from our proposed PFSTS clustering II are better in both cases. The widths chosen by our algorithm are more accurate, as a result from the use of MDL principle where we can make higher quality decisions in selecting proper widths, and no matter how different the subsequences' widths in time series are, our algorithm can handle it well. As seen in Fig. 9 that the datasets we used consists of very large difference among different subsequences' widths.

5 Conclusion

In this work, we have demonstrated that the existing STS clustering algorithms cannot deal with the variable-width problem well since the variability is limited under some predefined parameters. Then, we propose an algorithm to resolve the problem based on our preliminary work on parameter-free STS clustering, with an addition of applying MDL principle and other techniques to further improve the clustering accuracy, validity, and width variability. The experiment results have shown that our proposed algorithm outperforms all other previously proposed approaches while maintaining its parameter-freeness property and producing more accurate clustering results, especially in the case where the variability in the subsequence widths is large.

References

1. Keogh, E.J., Lin, J., Truppel, W.: Clustering of Time Series Subsequences is Meaningless: Implications for Previous and Future Research. In: Proceedings of the IEEE International Conference on Data Mining (ICDM), pp. 115–122 (2003)
2. Das, G., Lin, K., Mannila, H., Renganathan, G., Smyth, P.: Rule Discovery from Time Series. In: Proceedings of the 3rd Knowledge Discovery and Data Mining (KDD) (1998)
3. Zakaria, J., Mueen, A., Keogh, E.: Clustering Time Series Using Unsupervised-Shapelets. In: Proceedings of the IEEE International Conference on Data Mining (ICDM), pp. 785–794 (2012)
4. Barron, A., Rissanen, J., Yu, B.: The minimum description length principle in coding and modeling. IEEE Transactions on Information Theory 44(6), 2743–2760 (1998)
5. Fu, T.: A review on time series data mining. Engineering Applications of Artificial Intelligence 24, 164–181 (2011)
6. Rakthanmanon, T., Keogh, E.J., Lonardi, S., Evans, S.: Time Series Epenthesis: Clustering Time Series Streams Requires Ignoring. In: Proceedings of the 11th IEEE International Conference on Data Mining (ICDM), pp. 547–556 (2011)
7. Mueen, A., Keogh, E.J., Zhu, Q., Cash, S., Westover, M.B.: Exact Discovery of Time Series Motifs. In: Proceedings of the SIAM International Conference on Data Mining, pp. 473–484 (2009)
8. Rodpongpun, S., Niennattrakul, V., Ratanamahatana, C.A.: Selective Subsequence Time Series clustering. Knowledge-Based Systems 35, 361–368 (2012)
9. Keogh, E.J., Xi, X., Wei, L., Ratanamahatana, C.A., The, U.C.R.: The UCR time series classification/clustering homepage (2008),
`http://www.cs.ucr.edu/~eamonn/time_series_dat/`

10. Cotofrei, P., Stoffel, K.: Classification Rules + Time = Temporal Rules. In: Sloot, P.M.A., Tan, C.J.K., Dongarra, J., Hoekstra, A.G. (eds.) ICCS-ComputSci 2002, Part I. LNCS, vol. 2329, pp. 572–581. Springer, Heidelberg (2002)
11. Yingchareonthawornchai, S., Sivaraks,Rodpongpun, S., Ratanamahatana, C.A.: The Proper Length Motif Discovery Algorithm. In: Proceedings of the 16th International Computer Science and Engineering Conference (ICSEC 2012), Chonburi, Thailand (2012)
12. Madicar, N., Sivaraks, H., Rodpongpun, S., Ratanamahatana, C.A.: Parameter-free subsequences time series clustering with various-width clusters. In: 2013 5th International Conference on Knowledge and Smart Technology (KST), pp. 150–155 (2013)
13. Niennattrakul, V., Wanichsan, D., Ratanamahatana, C.A.: Accurate Subsequence Matching on Data Stream under Time Warping Distance. In: Theeramunkong, T., Nattee, C., Adeodato, P.J.L., Chawla, N., Christen, P., Lenca, P., Poon, J., Williams, G. (eds.) New Frontiers in Applied Data Mining. LNCS, vol. 5669, pp. 156–167. Springer, Heidelberg (2010)
14. Wang, S., Gan, W., Li, D., Li, D.: Data Field for Hierarchical Clustering. International Journal of Data Warehousing and Mining archive (IJDWM) 7(4), 43–63 (2011)

An Optimized Classification Approach Based on Genetic Algorithms Principle

Ines Bouzouita

Computer Science Department,
Faculty of Sciences of Tunis, 1060 Tunis, Tunisia
ines.bouzouita@yahoo.fr

Abstract. In this paper, we address the problem of generating relevant classification rules. Within this framework we are interested in rules of the form $a_1 \wedge a_2 \ldots \wedge a_n \Rightarrow b$ which allow us to propose a new approach based on the cover set and genetic algorithms principle. This approach allows obtaining frequent and rare rules while avoiding making a breadth search. It is an improvement of AFORTIORI approach. Moreover, our proposed algorithm can extract the classifier using a clustering for the attributes which allows to minimize the processing of the classifier building.

Keywords: Clustering, Cover Set, Associative Classification, Rare Classification Rules.

1 Introduction

In the last few years, a new approach that integrates association rule mining and classification called associative classification has been proposed. Several accurate and effective classifiers based on associative classification have been presented in last few years and are interested in the extraction of valid classification rules, with sufficient values of support and confidence [1–6]. However, in some applications, support threshold limits the research space and so removes rare and interesting rules. Indeed, such rules may predict rare genetic diseases or medical diagnosis that could be found by decreasing the minimum support threshold, which drastically increases the runtime of the pattern extraction algorithm.

In this paper, we address the problem of generating relevant classification rules. Within this framework we are interested in rules of the form $a_1 \wedge a_2 \ldots \wedge a_n \Rightarrow b$ which allow us to propose an optimized approach based on the cover set classical algorithm. This approach allows obtaining frequent and rare rules while avoiding making a breadth search. We introduce an associative classification algorithm, an improvement of AFORTIORI approach. Moreover, our proposed algorithm extracts the classifier using a clustering for the attributes which allows to minimize the processing of the classifier building.

The remainder of the paper is organized as follows. Section 2 introduces our proposed approach, where details about classification rules discovery and experimental results are given. Finally, section 3 concludes this paper.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_41, © Springer International Publishing Switzerland 2014

2 Optimized Classification Approach

As already mentioned, the extraction of rare classification rules may be particularly useful in biology and medicine. Let us take an example from the field of pharmacovigilance, *i.e.*, a field of pharmacology dedicated to the detection, survey and study of adverse drug effects [7]. Given a database of adverse drug effects, rare classification rules extraction enables a formal way of associating drugs with adverse effects, i.e. finding cases where a drug had fatal or undesired effects on patients. In this way, a frequent classification rule such as drug $A \rightarrow Cl1$, where $Cl1$ is a label describing a kind of desirable effect, means that this classification rule describes an expected and right way of acting for a drug. By contrast, a rare classification rule such as drug $A \rightarrow Cl2$ may be interpreted as the fact that $Cl2$ describes an abnormal way of acting for a drug, possibly leading to an undesirable effect. Thus, searching for adverse effects for a drug may be stated as a search for rare classification rules in a database.

In this section, we propose an optimized AC method of AFORTIORI [8] that extracts relevant classification rules from a learning data set. In the following, we will present and explain in detail the proposed approach.

The intuition behind AFORTIORI method is that AC algorithms [1–3] utilize the frequent itemset strategy as exemplified by the Apriori algorithm. The frequent itemset strategy first discovers all frequent itemsets, *i.e.*, whose support exceeds a user defined threshold minimum support. Associative classification rules are then generated from these frequent itemsets. These approaches are efficient if there are relatively few frequent itemsets. It is, however, subject to a number of limitations:

1. Associations with support lower than the nominated minimum support will not be discovered. Infrequent itemsets may actually be especially interesting for some applications. In many applications high value transactions are likely to be both relatively infrequent and of great interest.

2. Even if there is a natural lower bound on support the analyst may not be able to identify it. If minimum support is set too high then important associations will be overlooked. If it is set too low then processing may become infeasible. There is no means of determining, even after an analysis has been completed, whether it may have overlooked important associations due to the lower bound on support being set too high.

In this following, we will recall some logic basic notions necessary for cover set algorithm principle comprehension.

2.1 Preliminaries

A formal context is a triplet $\mathcal{K} = (E, A, R)$, where E represents a finite set of examples, A is a finite set of attributes and R is a binary (incidence) relation.

For $A_1 \subseteq A$, we have e(A_1) a set of $e_j \in E$ / $\forall a_k \in A_1, R(e_j, a_k)$

Within the framework of Formal Concept Analysis (FCA): $A_1 \subseteq A$ corresponds to conjunctive formula $a_1 \wedge a_2 \ldots \wedge a_n$ with $a_j \in A_1$.

To compute the set of example verifying A, *e(A)*, i.e., e($a_1 \wedge a_2 \ldots \wedge a_n$) we compute e($a_1$) ∩e($a_2$) ∩... e($a_n$).

For a disjunctive formula e($a_1 \vee a_2 \ldots \vee a_n$), i.e, examples verifying at least one attribute, e($a_1 \vee a_2 \ldots \vee a_n$)=e($a_1$) ∪e($a_2$) ∪... e($a_n$)

We can extend this to any logical formula considering that we search all the elements of E checking it. We extract from (E, A, R), rules of the form $a_1 \wedge a_2 \ldots \wedge a_n \Rightarrow b$ with $b \in A$. We can extend this to $a_1 \wedge a_2 \ldots \wedge a_n \Rightarrow B$ where B is a logical formula.

Within the framework of FCA and datamining: $a_1 \wedge a_2 \ldots \wedge a_n \Rightarrow b$ is true if and only if $e(a_1 \wedge a_2 \ldots \wedge a_n) \subseteq e(b)$. We know, from [9], that a rule $B \Rightarrow b$ in a context (E,A,R) is true if and only if in the complementary context (E,A,R') each object in $e(b)$ has, at least, one attribute a in A. This proposal can be reformulated in the following one as follows:

Proposition 1. *For a context* (E, A, R)*,* $a_1 \wedge \ldots \wedge a_n \Rightarrow b \iff e(\neg a_1) \cup \ldots \cup e(\neg a_n) \cup e(b) = E$*.*

Proof. For a context (E, A, R),

$a_1 \wedge \ldots \wedge \mathrm{a}_n \Rightarrow b \iff (\neg a_1 \vee \ldots \vee \neg a_n) \vee b$ is true (logic).

We know 1) $a_1 \wedge \ldots \wedge \mathrm{a}_n \Rightarrow b \iff e(a_1) \cap \ldots e(a_n) \subseteq e(b)$ and 2) $e(a) = E - e(\neg a)$

Then, we have $(\neg a_1 \vee \ldots \vee \neg a_n) \vee b$ is true $\iff e(\neg a_1 \vee \ldots \vee \neg a_n) \cup e(b) = E$

So, the research of rules $a_1 \wedge \ldots \wedge \mathrm{a}_n \Rightarrow b$ for a context (E, A, R) is equivalent with a classical cover set problem [10].

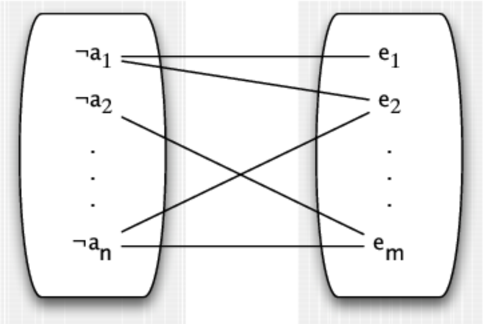


Fig. 1. Bipartite representation

2.2 Cover Set Problem

Given a set $\mathcal{N}$ and a family $\mathcal{S}$ of subsets of $\mathcal{N}$, a cover is a subfamily $\mathcal{C} \subseteq \mathcal{S}$ of sets whose union is $\mathcal{N}$. It is easy to see this by observing that an instance of cover set can be viewed as an arbitrary bipartite graph as shown by figure 1, with sets represented by vertices on the left, the universe represented by vertices on the right, and edges representing the inclusion of elements in sets. The task is then to find a subset of left-vertices which covers all of the right-vertices.

Proposition 2. *For a context (E, A, R) and an attribute b in A. The search of all rules: $a_1\wedge ... \wedge a_n \Rightarrow b$ with a_i in A, is equivalent with the search of all cover sets of $E - e(b)$ using subsets $e(\neg a_i)$ in A and $e(\neg a_i)$ is $E - e(a_i)$.*

Proof Comes directly from the property 1 $a_1\wedge ... \wedge a_n \Rightarrow b \Longleftrightarrow e(\neg a_1) \cup ... \cup e(\neg a_n) \cup e(b) = E$ since, for each rule, we have an union of subsets which cover E and for each cover set a rule.

The set cover problem is a really important one. There are several papers and algorithms treating this issue [9].

Our work is motivated by the long-standing open question of devising an efficient algorithm for finding rules with low support. A particularly relevant field for rare item sets and rare associative classification rules is medical diagnosis. For example it may be that in a large group of patients diagnosed with the same sickness, a few patients exhibit unusual symptoms. It is important for the doctor to take this fact into consideration.

In this section, we introduce an associative classification algorithm, an improvement of AFORTIORI approach.

We have adopted genetic algorithms principle to minimize the search space exploration and thus reduce building classifier runtime. Indeed, the algorithm 1 appeals a procedure given by a greedy algorithm 2 based on the following principle. An expert of the domain of the application set a number of attributes k which are randomly chosen among the list of the candidates. This set of attributes corresponds to a part of the premise of a rule, called granulate, which will be completed to cover the examples of the context in question. Then, the granulate is built by crossing premises of two existing rules. We repeat the process until reach the number of rules set from the beginning by an expert of the field of research of the application.

Moreover, our algorithm extracts the classifier using a clustering for the attributes which allows to minimize the processing of the classifier building.

The algorithm 2 uses a procedure outlined by Algorithm 3 which is a heuristic based on an evaluation function. Algorithm 2 explores the search space in a depth-first manner, for this it uses an evaluation function for the choice of next attribute to be treated. Before each recursive call, it lists all attributes covering the same examples in a context in the same group G (i) using the method Create-attributes clusters outlined by Algorithm 2. So, it is sufficient to treat a single attribute of the cluster that at the end of treatment we obtain rules with a premise combination of cluster G (i) knowing that each cluster forms a disjunction of attributes. Thus, unlike classical AFORTIORI approach, the proposed version permits to minimize the processing step by creating attributes clusters while building the classifier.

The intuition behind our approach is that both frequent and rare associative classification rules could be of important interest for many real applications area. Unlike previous AC approaches, our new approach permits:

- to explore the search space in a depth−first manner.
- to extract both frequent and rare classification rules by using an evaluation function.

```
Data: Context: (E,A,R), Attribute class
Results: classifier
nbg: generation number;
x: rules number;
Build sub-complementary context (E',A',R') of (E,A,R)
While (nbr-generation< nbg) do
    While (nbr-regle $\leq$ x)
    do
        pathinit= choixalea(la,k)
        classifier$\leftarrow$Search((E,A,R),(E',A',R'),pathInit,res,class)
        Apply proposed genetic opearators
    End
End
```

Algorithm 1. Our proposed approach based on genetic algorithms principle

```
Search
Data:  Context: (E,A,R); Sub complementary context: (E',A',R'); List of
       attributes: pathinit, Res; Attribute b
Results:  List of rules:Res
Begin
    If E' is empty then
        If support $\overline{pathinit}$ >0 then
            add rule $\overline{pathinit}$$\rightarrow$ b to Res
        If A is empty then
            return Res (Backtrack)
    Res $\leftarrow$ pathInit;
    Rest-attribut=LA-pathInit;
    Evaluate each attribute in A' using the evaluation function
    Evaluate each attribute in "Rest-attribut" using the cover set principle
    Order the attributes in a list "Rest-attribut"
    While Rest-attribut is not empty do
        Take the first element x of Rest-attribut
        Create the new context (E",A",R") from (E',A',R') where A"$\leftarrow$A'- x,
        and E"$\leftarrow$E'-{e $\in$ e(x)}
        Create attributes clusters((E",A",R"))
        Res $\leftarrow$ Res $\cup$ Search((E,A,R),(E",A",R"), pathinit $\cup x$,Res,b)
        remove attribute x from Rest-attribut
    return Res
End
```

Algorithm 2. Optimized Backtrack search

In fact, our proposed algorithm traverses the space search in a depth-first manner to generate rules with small premises which is an advantage for the classification framework when the rules imply the same class. For example, let us consider two rules R_1: $a \wedge b \wedge c \wedge$d $\Rightarrow$`class` and R_2: $b \wedge c$ $\Rightarrow$`class`. R_1 and R_2 have the same attribute conclusion. R_2 is considered to be more interesting than R_1, since it

```
Create attributes clusters
Data: Context: (E,A,R)
Results: Context: (E,A,R)
Begin
    While $|A| > 1$ do
        x=1
        for $i=2$ to $|A|$ do
            If $E(A(x))=E(A(i))$ then
                Then G(j)$\leftarrow$ G(j)$\cup$A(i)//Grouping attributes covering the same
                examples
                A-A(i)
        end for
        G(j)$\leftarrow$ G(j)$\cup$A(x)
        A-A(x)
        If $|A| = 1$ then
            G(j+1)=A(1)
    end while
End
```

Algorithm 3. Create attributes clusters

	a	*b*	*c*	*d*	*e*	*f*	*class*
e1	1	0	0	1	0	0	0
e2	0	1	0	0	1	0	1
e3	0	0	1	0	0	1	1
e4	0	0	1	0	1	0	0
e5	1	0	0	1	0	0	1

Fig. 2. Example

is needless to satisfy the properties $a \wedge d$ to choose the class `class`. Hence, R_2 implies less constraints and can match more objects of a given population than R_1. In fact, such set of rules is smaller than the number of all the classification rules since we eliminate redundant ones. Moreover their use is beneficial for classifying new objects.

2.3 Evaluation Function

In this paragraph, we define an evaluation function for our proposed algorithm. For this, we use a classical evaluation function used in the greedy algorithm for the set covering optimization problem. In the set covering decision problem, the input is a pair ($\mathcal{N}$,$\mathcal{S}$) and an integer k; the question is whether there is a set covering of size k or less. In the set covering optimization problem, the input is a pair ($\mathcal{N}$,$\mathcal{S}$), and the task is to find a set covering which uses the fewest sets. This problem is NP-hard but there is an approximation algorithm for this problem.

Our Greedy algorithm 4 works by picking, at each stage, the a which e(a) covers the most remaining uncovered elements.

```
Data: Context: (E,A,R') (complementary context)
Results: C
Begin
    U = E
    C =∅
    While (U ≠ ∅)and (C ≠ A) do
        Select an attribute a in A - C, with e(a) cover the maximum number
        of uncovered attributes
        U = U - e(a)
        Add a to the Cover C being constructed
    return C
End
```

Algorithm 4. Greedy Cover Set algorithm

The Greedy Cover Set algorithm 4 works as follows: set E (examples) is the main set and set A (attributes) contains a defined collection of Subsets e(a) . The set U contains at each stage, the set of remaining uncovered elements. The set C contains the cover being constructed. In a Greedy approach, always, a subset e(a) is chosen that covers as many uncovered elements as possible. After that e(a) is selected, the elements are removed from U and a is placed in C. When the algorithm terminates the set C contains, if there is one, a subfamily of A such that the Union of e(ak) in U covers E.

For our problem, this method gives the following results:

Proposition 3. *For a context (E, A, R), we can prove in polynomial time whenever there is a rule of the form $a_1\wedge ... \wedge a_n \Rightarrow b$.*

Proof. We know, from proposition 1, that $a_1\wedge ... \wedge \mathrm{a}_n \Rightarrow b \Longleftrightarrow e(\neg a_1) \cup ... \cup e(\neg a_n) \cup e(b) = E$

The search for $e(\neg a_1) \cup ... \cup e(\neg a_n) \cup e(b) = E$ is the classical form of a set covering problem.

More formally, given a universe U and a family S of subsets of U, a cover is a subfamily $C \subseteq S$ of sets whose union is E [10].

In our case, the universe is E and the family of subsets S are $e(\neg a_i) = E - e(a_i) \subseteq E$ and $e(b) \subseteq E$.

Using the classical greedy algorithm [10] for the cover set problem on E and S, we can find a rule in polynomial time of the form $a_1\wedge ... \wedge \mathrm{a}_n \Rightarrow b$ if it exists.

Proposition 4. *For a context (E, A, R), we can prove in polynomial time if there is a rule of the form $a_1\wedge ... \wedge a_n \Rightarrow b$ with support $\neq 0$.*

Proof. For $e \in e(b)$ we search for a rule $a_1\wedge ... \wedge \mathrm{a}_n \Rightarrow b$, with $\forall a_i \in \{a_1 \dots a_n\}, e \in e(a_i)$. The search for this rule is polynomial (proposition 3), by construction, the support of this rule is $\geqslant 1$.

We can do that for all $e \in e(b)$, and then prove in polynomial time that there is a rule of the form $a_1\wedge ... \wedge \mathrm{a}_n \Rightarrow b$ with support $\neq 0$. The complexity of this method is $O(\mid E \mid *CoverSet)$.

Considering the context given by Figure 2, our proposed algorithm, using the cover evaluation function, generates the following rules :
R_1: b ⇒ class=1
R_2: f ⇒ class=1

Experiments were done on medical data set SPECT Heart taken from UCI Machine Learning Repository$^{(1)}$. The data set describes diagnosing of cardiac Single Proton Emission Computed Tomography (SPECT) images.

Rare rules classify correctly certain objects. Indeed, the omission of such rules by studied popular AC approaches -because of the cost of their generation- makes the accuracy lower than that of our approach classifier which contains rare rules. This justifies the usefulness of the presence of such rules in the classifier for studied bases. In our experiments, we study the accuracy values given by AC approaches on Spect Heart data set. Results show that for the studied medical data set rare rules give good accuracy results.In fact, only our approach gives an accuracy value of 79.0%.

Moreover, this approach, thanks to the adopted research method does not generate a large patterns number and thus avoids the problems encountered by classical AC approaches.

3 Conclusion

In this paper, we introduced a classification approach based on genetic algorithms principle that extract relevant classification rules while exploring the search space in a depth−first manner. Moreover, it can extract the classifier using a clustering for the attributes which allows to minimize the processing of the classifier building.

References

1. Liu, B., Hsu, W., Ma, Y.: Integrating classification and association rule mining. In: Knowledge Discovery and Data Mining, pp. 80–86 (1998)
2. Antonie, M., Zaiane, O.: Text Document Categorization by Term Association. In: Proceedings of the IEEE International Conference on Data Mining (ICDM 2002), Maebashi City, Japan, pp. 19–26 (2002)
3. Antonie, M., Zaiane, O.: Classifying Text Documents by Associating Terms with Text Categories. In: Proceedings of the Thirteenth Austral-Asian Database Conference (ADC 2002), Melbourne, Australia (2002)
4. Wang, J., Karypis, G.: HARMONY: Efficiently mining the best rules for classification. In: Proceedings of the International Conference of Data Mining, pp. 205–216 (2005)
5. Bouzouita, I., Elloumi, S., Yahia, S.B.: Garc: a new associative classification approach. In: Tjoa, A.M., Trujillo, J. (eds.) DaWaK 2006. LNCS, vol. 4081, pp. 554–565. Springer, Heidelberg (2006)

[1] *Available at* `http://www.ics.uci.edu/~mlearn/MLRepository.html`

6. Bouzouita, I., Elloumi, S.: Integrated generic association rules based classifier. In: Proceedings of Eighteenth International Workshop on Dtabase and Expert Systems Applications (DEXA 2007), Regensburg, Germany, pp. 514–518 (2007)
7. Szathmary, L., Napoli, A., Valtchev, P.: Towards rare itemset mining. In: Proceedings of the 19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2007), vol. 1, pp. 305–312. IEEE Computer Society, Washington, DC (2007)
8. Bouzouita, I., Liquiere, M., Elloumi, S.: Garc: Afortiori: a new associative classification approach based on cover set algorithm. In: Wolff, K.E., Rudolph, S., Ferre, S. (eds.) Proceedings of 7th International Conference on Formal Concept Analysis Darmstadt, Germany, May, pp. 51–62 (May 2009)
9. Ganter, B., Wille, R.: Formal Concept Analysis. Springer (1999)
10. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, 2nd edn. MIT Press and McGraw-Hill (2001)

Comparative Performance Analysis of Negative Selection Algorithm with Immune and Classification Algorithms

Ayodele Lasisi[1], Rozaida Ghazali[1], and Tutut Herawan[2]

[1] Faculty of Computer Science and Information Technology
Universiti Tun Hussein Onn Malaysia
86400, Parit Raja, Batu Pahat, Johor, Malaysia
lasisiayodele@yahoo.com, rozaida@uthm.edu.my
[2] Faculty of Computer Science and Information Technology
University of Malaya, 50603, Kuala Lumpur, Malaysia
tutut@um.edu.my

Abstract. The ability of Negative Selection Algorithm (NSA) to solve a number of anomaly detection problems has proved to be effective. This paper thus presents an experimental study of negative selection algorithm with some classification algorithms. The purpose is to ascertain their efficiency rates in accurately detecting abnormalities in a system when tested with well-known datasets. Negative selection algorithm with some selected immune and classifier algorithms are used for experimentation and analysis. Three different datasets have been acquired for this task and a comparison performance executed. The empirical results illustrates that the artificial immune system of negative selection algorithm can achieve highest detection and lowest false alarm. Thus, it signifies the suitability and potentiality of NSA for discovering unusual changes in normal behavioral flow.

Keywords: anomaly detection, classification algorithm, data representation, negative selection algorithm.

1 Introduction

The emergence of Artificial Immune System (AIS) which began with the works of Forrest and her group [1] in 1994 by proposing and developing the Negative Selection Algorithm (NSA) opened the door as an important addition within the confines of anomaly detection. The biological process of *negative selection*, which laid the foundation for the abstraction of NSA algorithm, is attributed to specialized white blood cells, called T-cells developed in the bone marrow inhibiting receptors that undergoe a pseudo-random generation procedure. These T-cells are further exposed to $self$ cells in the thymus, and there is an elimination action taken when a reaction occurs between the the T-cells and the $self$ cells with those not reacting being retained and granted permission to leave the

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_42, © Springer International Publishing Switzerland 2014

thymus into maturation stage. At this stage, they are now fully integrated in the immune system surveillance structure of intuders into the body.

The task of anomaly detection, a two-class classification problem that classifies an element as *normal* or *abnormal* within a given feature space, can be considered as analogous to the immunity of biological immune system [2]. Their targeted aim is to detect abnormal behaviours of system that contradicts to the normal functioning of the system [3, 4]. Varieties of classification-anomaly based techniques exist in literature [5] ranging from neural-network based to rule-based. These classification-anomaly based methods establishes a balanced platform for comparison with the immune algorithm of negative selection algorithm, clonal selection algorithm, artificial immune recognition system (airs), and Immunos-81 variants. The purpose of venturing into algorithmic comparison as will be highlighted in this paper, is to weigh the performances in terms of detection rate and false alarm rate, which will in turn give us a clearer picture of their capacities when anomaly detection is concerned.

The structure of this paper is as follows. In Section 2, an overview of what anomaly detection is all about is presented. This is followed by a brief insight into classification algorithms in Section 3. Negative Selection Algorithm and further exploration of the two data representation types constitutes Section 4. Our experimental results and discussions of these results sums up Section 5. The paper concludes with Section 6.

2 Anomaly Detection

In comprehending the concept behind anomaly detection, the term *anomaly* must be clearly defined. An anomaly, also referred to or used interchangeably depending on the application area as outliers, aberrations, exceptions, or peculiarities, is defined as patterns behaving differently to the normal behavioral flow [5]. Three basic types of anomalies exist namely, point anomalies (single data instance deviation), collective anomalies (deviation of group data instances), and contextual anomalies (context of deviation occurrence). Thus, anomaly detection is the process of identifying or recognizing abnormal behavioral changes in data [5, 6]. It will be of interest to know that anomaly detection has been in existence since the nineteenth century [7]. Improvement and advances in technological approach to effectively detect anomalies has been the distinguishing factor, and this is echoed by Vasarhelyi and Issa [8]. Computer security, medical, computer vision, general purpose data analysis and mining, and sensor network are some of the application areas of anomaly detection [9].

An anomaly detection method is considered good irrespective of the domain, based on the following three criteria [10]: (1) accurately locating and differentiating anomalies from normal behavior, (2) robustness in terms of sensitivity to parameter settings and changes of patterns in datasets, and (3) limited resources required. The training data to be used for modeling the system is of great importance as it aids in selecting the appropriate anomaly detection techniques which are supervised, semi-supervised, and unsupervised learning techniques.

Two phases, training phase and testing phase, make up the supervised technique where both malicious and benign data are applied. The semi-supervised technique on the other hand requires only the normal data with the usage of a training phase. However, unsupervised technique is devoid of a training phase with all the data present prior to initializing the algorithm [11].

3 Classification Algorithm

Classification is the process of generalizing data according to different instances [12] and predicting a certain outcome based on a well-known given input [13]. Various classification techniques are available for solving classification problems ranging from statistical methods, decision trees, neural networks, rule-based methods [14, 15]. The training data serves as input for the classification and rule based algorithm to begin their tasks for which the target values are known. With each of the algorithms having different strengths and weaknesses, they possess the ability to find relations between the predictor attributes' values and target attributes' values in the training data [16]. The selected algorithms for use in this paper are the popularly known and a few artificial immune system classifiers which will be brought to light in the later section of this paper.

4 Negative Selection Algorithm

The human immune system process called *negative selection* gave rise to one of the earliest algorithms in the artificial imunne system domain. The theoretical concept of negative selection is well rooted in central tolerance, an immune mechanism for self-tolerance averting autoimmunity (reacting to self antigens) [17]. T-cells, a special kind of white blood cell called lymphocytes, are generated from the bone marrow and equipped with receptors which takes upon themselves the task of identifying and recognizing specific molecular patterns. The receptors of T-cells are generated in a pseudo-random manner, and are exposed to normal proteins which resides in the thymus of the host body. The reaction of the T-cells with the proteins causes an elimination of the T-cells, and only those which do not react are allowed to migrate from the thymus, causing them to mature and be fully integrated in the immune system [18]. Based on the negative selection principle, Forrest et al. [1] proposed and developed the Negative Selection Algorithm (NSA) for detection applications. Two data representations of NSA are the strings (or binary) negative selection algorithm, and real-valued negative selection algorithm [19].

4.1 Strings Representation of Negative Selection Algorithm

The processes of negative selection made it well suited for computer and network security, and in 1994, [1] proposed a Negative Selection Algorithm (NSA) for discriminating between self and non-self in a computer. It can be applied to virus detection, image inspection and segmentation, and also hardware tolerance [20]. Steps in NSA execution is summarized as follows [21]:

Given a universe U which contains all unique bit-strings of length l, self set $S \subset U$ and non-self set $N \subset U$, where

$$U = S \subset U \qquad \text{and} \qquad S \cap N = \emptyset$$

1.˙ Define self as a set S of bit-strings of length l in U.
2. Generate a set D of detectors, such that each fails to match any bit-string in S.
3. Monitor S for changes by continually matching the detectors in D against S.

Since the first implementation of NSA which uses the Exhaustive Detector Generating Algorithm (EDGA) incorporated into the works of Forrest et al. [1], different variations of the algorithm using strings representation have been reported in literature [22]. This stems from limitations of the original NSA, as the time in generating valid detectors increases exponentially with the size of the self strings (time and space complexity) [23]. Thus, larger number of detectors are been generated. As a measure against the time and space consuming factors, D'Haeseleer et al. [24] developed the linear and greedy detector generation algorithms, with both operating in linear time with respect to the size of the self and detector sets, and greedy algorithm was reported to dissolve the problem with a new collection of generated detectors. Also, Wierzchon [25] introduced binary template with no intention of decreasing the time but rather generating efficient non-redundant detectors. NSMutation proposed in [26] is a modified version of EDGA using somatic hypermutation, and Ayara et al. [27] made a performance comparison of the different strings detector generation algorithms and results showed that NSMutation is more extensible. Still, the strings representation suffers greatly in dealing with real world applications which basically inherits the use of real-valued data.

4.2 Real-Valued Representation of Negative Selection Algorithm

The proposition by Gonzalez et al. [15] employs the use of real-valued data as against strings representation, in an effort to deal with the issues posed by strings (or binary) negative selection algorithm, and termed it Real-Valued Negative Selection Algorithm (RNSA). The algorithm distributes the detectors in the $nonself$ space based on heuristic to optimally maximize the coverage area. Among the advantages of real-valued representation which is a high level representation stated in [28, 29] are increased expressiveness, possibility of extracting high-level knowledge from the generated detectors, and improved scalability in certain cases. The algorithm adopts n-dimensional vectors in real space $[0, 1]^n$ to encode antigens and antibodies, and Euclidean distance to calculate the affinities between them.

Although the RNSA is characterized by the three basic steps in [21], the algorithmic description with additional components in evolving the detectors to cover the *nonself* space is found in [15]. This is performed through an iterative process with the aim of:

- Moving the detectors away from the *self* set, and
- Maximizing the coverage space of *nonself* by keeping the detectors apart

The detectors of the RNSA is fixed and chosen beforehand, and in an effort to dynamically choose the detectors, [30] proposed an improved version of RNSA called Variable-Sized Detectors (V-Detectors). The detectors of V-Detectors terminates training stage when enough coverage has been achieved. For the purpose of study in this paper, the RNSA [15] will be used to have a balanced performance comparison with the chosen classification algorithms.

5 Experimental Results and Analysis

Experiments are performed to compare the performance of Negative Selection Algorithm with some classification algorithms. These algorithms are implemented using both MATrix LABoratory (MATLAB) and Waikato Environment for Knowledge Analysis (WEKA). The selected classifiers from WEKA toolbox are the immune algorithms namely AIRS1 [31, 32], AIRS2 [31, 33], AIRS2Parallel [31, 34], CLONALG [35], Immunos1 [36], Immunos2 [36], and Immunos99 [36]. Selecting from different categories, the standard classification algorithms adopted for use in the experimental study includes Naive Bayes (NB) from bayesian category, Multilayerperceptron (MLP) from neural network, Sequential Minimal Optimization (SMO) from support vector machines category, IBk from instance-based category, J48 from decision tree, and NNge from nearest neighbour category. Dataset have been retrieved from UCI Machine Learning Repository and are all real-valued data which suites well for implementation with negative selection algorithm, and they are Fisher's IRIS data, Balance-Scale (BS), and Lenses data.

The Fisher's IRIS data has largely been employed for use in discriminant analysis and cluster analysis. It is composed of three species of 50 samples each, *Iris Setosa*, *Iris Versicolor*, and *Iris Virginica* with four numeric features, *sepal length*, *sepal width*, *petal length*, and *petal width*. These features are measured in millimeters within an entire searching space of 4-dimensional hypercube $[0, 1]^4$. The Balance-Scale on the otherhand has 625 data instances with three classes of balance-scale *tip to the left*, *tip to the right*, and *balanced*, while Lenses comprises of 24 data instances and three classes as well. The searching space of both Balance-Scale and Lenses is a 4-dimensional hypercube $[0, 1]^4$.

In other to pass the datasets as input for NSA execution in MATLAB, each class is employed as the training data while the other classes becomes the testing data used to measure the performance of the detectors. For example, one of the species of the Fisher's IRIS data serves as training set, and the other two species are for testing. This process is performed for each class, and the Euclidean distance in (1) is integrated to measure the affinities between the detectors and real-valued coordinates.

$$D = \sqrt{\sum_{i=1}^{n}(d_i - x_i)^2} \tag{1}$$

where $d = d_1, d_2, \ldots, d_n$ are the detectors, $x = x_1, x_2, \ldots, x_n$ are the real-valued coordinates, and D is the distance. The parameters used by real-valued negative selection algorithm are: $r = 0.1$, $\eta_o = 0.005$, $t = 15$, and $\tau = 15$. The number of randomly generated detectors is 1000, and experiments were repeated 10 times with the average values recorded. The above parameters denotes:

- r: radius of detection
- η_o: initial value of the adaptation rate
- t: age of the detector
- τ: the decay rate

To assess the performance of the immune and classification algorithms in WEKA, a 10-fold cross validation is used for testing and evaluating. This process entails dividing the dataset into ten subsets of equal size, with nine subsets making up the training data, leaving the only remaining subset as the test data. Performance statistics are calculated across all 10 trials, and this provide a good platform to ascertain how well the classifiers perform on the various dataset.

5.1 Performance Metric Terms

As a measure for balanced performance comparison between real-valued negative selection algorithm, immune algorithms, and classification algorithms, the detection rate and false alarm rate described in (2) and (3) are used for evaluation.

$$DR = \frac{TP}{TP + FN} \tag{2}$$

$$FAR = \frac{FP}{FP + TN} \tag{3}$$

where TP is the number of $nonself$ elements identified as $nonself$; TN is the number of $self$ elements identified as $self$; FP is the number of $self$ elements identified as $nonself$; FN is the number of $nonself$ elements identified as $self$; DR is the detection rate; FAR is the false alarm rate.

5.2 Simulation Results

The simulation experiments were performed on 2.10 GHz Intel Pentium (R) Processor with 4GB of RAM. As earlier mentioned that MATLAB and WEKA

Table 1. Fisher's IRIS dataset performance result

Algorithm	Detection Rate %	False Alarm Rate %
AIRS1	95.33	2.3
AIRS2	95.33	2.3
AIRS2Parallel	96.0	2.0
CLONALG	95.33	2.3
Immunos1	97.33	1.3
Immunos2	97.33	1.3
Immunos99	96.67	1.7
Naive Bayes	96.0	2.0
Multilayerperceptron	97.33	1.3
SMO	96.0	2.0
IBk	95.33	2.3
J48	96.0	2.0
NNge	96.0	2.0
NSA	**96.73**	**0.53**

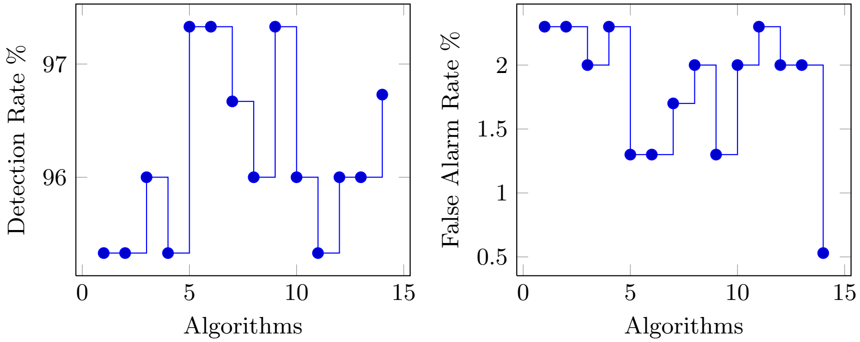


Fig. 1. Graph Plots for Detection Rates and False Alarm Rates (Fisher's IRIS)

have been adopted for use in the performance comparison of the algorithms. The results after series of experiments for each dataset is tabulated and graphed.

In the graph illustrations, the algorithms have been labelled from 1 to 14 with AIRS1 representing the least, followed by AIRS2, while NSA connotes the highest in the order shown from the tables. In Table 1, it can be revealed that Negative Selection Algorithm performed considerably well when compared with the other selected algorithms on the Fisher's IRIS dataset. With a detection rate of 96.73%, only Immunos1, Immunos2, and MLP with 97.33% each were better in performance. With respect to false alarm rate, NSA shows to be more effective with 0.53%. AIRS1, AIRS2, CLONALG, and IBk each produced the highest false alarm rate of 2.3%. The graph representation of the above explanation is reflected in Figure 1 for both the detection rate and false alarm rate.

The overall performance of the Balance-Scale dataset is presented in Table 2 below. The values of the respective algorithms have been adequately plotted in a graph as depicted in Figure 2. The superiority of NSA with detection rate of 98.02% can be confirmed as against the immune and classification algorithms. Majority of the algorithms has detection rate which falls within the 80% range, with only Naive Bayes and MLP that could boast of reaching the 90% range having values of 90.4% and 90.72% respectively. The lower false alarm rate reported by NSA which equals 0.99% superceeds the higher false alarm rate generated by others, with CLONALG having the highest rate of 19.6%.

Table 2. Balance-Scale dataset performance result

Algorithm	**Detection Rate %**	**False Alarm Rate %**
AIRS1	80.48	11.9
AIRS2	80.96	12.2
AIRS2Parallel	81.76	11.5
CLONALG	75.2	19.6
Immunos1	71.84	3.1
Immunos2	86.72	11.3
Immunos99	69.12	5.2
Naive Bayes	90.4	8.2
Multilayerperceptron	90.72	4.2
SMO	87.68	10.5
IBk	86.56	9.5
J48	76.64	17.3
NNge	81.92	10.8
NSA	**98.02**	**0.99**

For Lenses dataset presented in Table 3 and diagrammatically shown in Figure 3, when tested with NSA, yielded a 100% detection rate, and this proves to outclass the other algorithms in which SMO could only attain 87.5% rate. In the same vein, other algorithms produced higher false alarm rate as against low rate of NSA with 1.22% rate. In addition, when compared to the other datasets, lenses data gave the lowest detection rate of 45.83% generated by Immunos99. Following the step laid down by the detection rate, lenses data also produced the highest false alarm rate of 58.3% to the experimentation with other datasets.

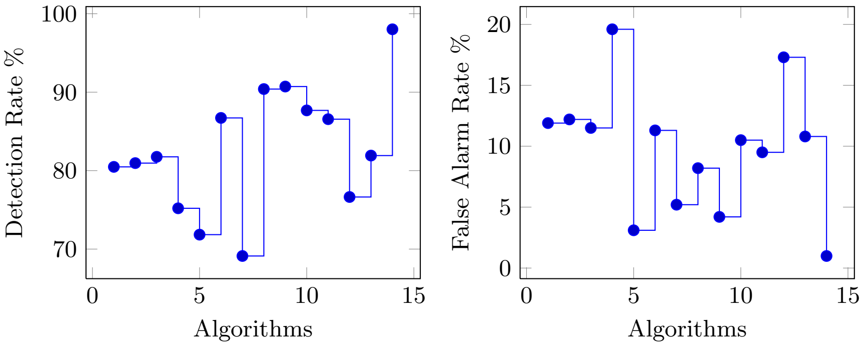


Fig. 2. Graph Plots for Detection Rates and False Alarm Rates (Balance-Scale)

Table 3. Lenses' dataset performance result

Algorithm	**Detection Rate %**	**False Alarm Rate %**
AIRS1	70.83	21.9
AIRS2	79.17	19.7
AIRS2Parallel	75.0	25.5
CLONALG	54.17	35.7
Immunos1	70.83	12.4
Immunos2	58.33	58.3
Immunos99	45.83	42.7
Naive Bayes	79.17	24.4
Multilayerperceptron	70.83	21.9
SMO	87.5	12.8
IBk	79.17	15.0
J48	66.67	23.0
NNge	70.83	21.9
NSA	**100.0**	**1.22**

Therefore, it can be said that NSA is well suited for anomaly detection which rest solely the idea as proposed by Forrest et al. [1], with severals experimental procedures reported in literatures over the years since its inception.

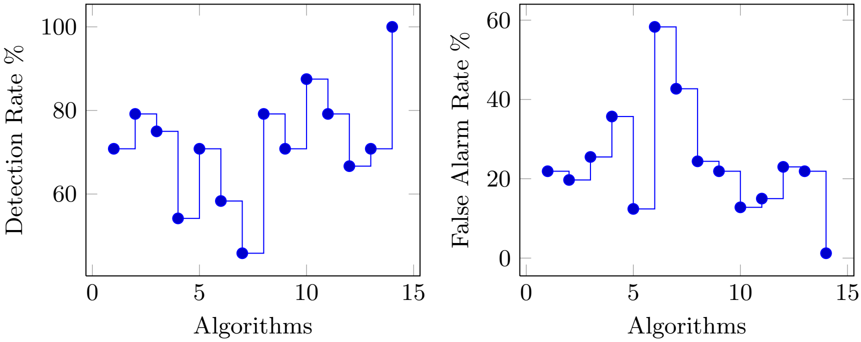


Fig. 3. Graph Plots for Detection Rates and False Alarm Rates (Lenses)

6 Conclusion

A comparative experimental study constitutes the backbone of this paper. Careful selection of classification-based anomaly techniques channel our objective for a proper comparison performance with immune algorithms. The focus is to determine how potent the algorithms could achieve their task when fed with data. The UCI repository provides with standard and benchmarked dataset, and three(3) of those dataset were used for this study. Experiments were carried out on a total of 14 classification and immune algorithms namely AIRS1, AIRS2, AIRS2Parallel, CLONALG, Immunos1, Immunos2, Immunos99, Naive Bayes, Multilayerperceptron, SMO, IBk, J48, NNge, and NSA for each dataset. The negative selection algorithm performed better than all the algorithms for two of the datasets with respect to rate of detection and false alarm, generating values of 98.02% and 0.99% for balance-scale data, also 100% and 1.22% for lenses data. On the remaining dataset (Fisher' IRIS), NSA proved its mettle with a detection rate of 96.73% (except for Immunos1,Immunos2, and MLP with 97.33%). With the verification of the anomaly detection potentials of NSA as reported in this study, and also with numerous improvements at enhancing its recognition qualities, further research will be directed at methods for boosting NSA algorithm.

Acknowledgements. As an expression of gratitude, the authors with deepest appreciation, would like to thank the Office for Research, Innovation, Commercialization and Consultancy Management (ORICC), Universiti Tun Hussein Onn Malaysia (UTHM) and Ministry of Higher Education (MOHE) for financially supporting this Research under the Exploratory Research Grant Scheme (ERGS), Vote No. 0882.

References

1. Forrest, S., Perelson, A.S., Allen, L., Cherukuri, R.: Self-nonself discrimination in a computer. In: Proceedings of the 1994 IEEE Computer Society Symposium on Research in Security and Privacy, pp. 202–212. IEEE (1994)
2. Patcha, A., Park, J.M.: An overview of anomaly detection techniques: Existing solutions and latest technological trends. Computer Networks 51(12), 3448–3470 (2007)
3. Boukerche, A., Machado, R.B., Jucá, K.R., Sobral, J.B.M., Notare, M.S.: An agent based and biological inspired real-time intrusion detection and security model for computer network operations. Computer Communications 30(13), 2649–2660 (2007)
4. Dasgupta, D., González, F.: An immunity-based technique to characterize intrusions in computer networks. IEEE Transactions on Evolutionary Computation 6(3), 281–291 (2002)
5. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. ACM Computing Surveys (CSUR) 41(3), 15 (2009)
6. Tamberi, F.: Anomaly detection (2007)
7. Edgeworth, F.: On discordant observations. The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 23(143), 364–375 (1887)
8. Vasarhelyi, M.A., Issa, H.: Application of anomaly detection techniques to identify fraudulent refunds (2011)
9. Song, X., Wu, M., Jermaine, C., Ranka, S.: Conditional anomaly detection. IEEE Transactions on Knowledge and Data Engineering 19(5), 631–645 (2007)
10. Yao, Y., Sharma, A., Golubchik, L., Govindan, R.: Online anomaly detection for sensor systems: A simple and efficient approach. Performance Evaluation 67(11), 1059–1075 (2010)
11. Amer, M., Abdennadher, S.: Comparison of unsupervised anomaly detection techniques. PhD thesis, Bachelor's Thesis 2011 (2011), `http://www.madm.eu/_media/theses/thesis-amer.pdf`
12. Kumar, R., Verma, R.: Classification algorithms for data mining: A survey. International Journal of Innovations in Engineering and Technology (IJIET) (2012)
13. Kilany, R.M.: Efficient classification and prediction algorithms for biomedical information (2013)
14. Weiss, S.M., Kulikowski, C.A.: Computer systems that learn: Classification and prediction methods from statistics, neural nets, machine learning and expert systems (1991)
15. Gonzalez, F., Dasgupta, D., Kozma, R.: Combining negative selection and classification techniques for anomaly detection. In: Proceedings of the 2002 Congress on Evolutionary Computation, CEC 2002, vol. 1, pp. 705–710. IEEE (2002)
16. Rao, K.H., Srinivas, G., Damodhar, A., Krishna, M.V.: Implementation of anomaly detection technique using machine learning algorithms. International Journal of Computer Science and Telecommunications, ISSN 2047–3338
17. Lederberg, J.: Genes and antibodies do antigens bear instructions for antibody specificity or do they select cell lines that arise by mutation? Science 129(3364), 1649–1653 (1959)
18. Textor, J.: A comparative study of negative selection based anomaly detection in sequence data. In: Coello Coello, C.A., Greensmith, J., Krasnogor, N., Liò, P., Nicosia, G., Pavone, M. (eds.) ICARIS 2012. LNCS, vol. 7597, pp. 28–41. Springer, Heidelberg (2012)

19. Lasisi, A., Ghazali, R., Herawan, T.: Negative selection algorithm: A survey on the epistemology of generating detectors. In: Herawan, T., Deris, M.M., Abawajy, J. (eds.) Proceedings of the First International Conference on Advanced Data and Information Engineering (DaEng 2013). LNEE, vol. 285, pp. 167–176. Springer, Heidelberg (2014)
20. Hofmeyr, S.A., Forrest, S.: Architecture for an artificial immune system. Evolutionary Computation 8(4), 443–473 (2000)
21. Stibor, T., Timmis, J., Eckert, C.: The link between r-contiguous detectors and k-cnf satisfiability. In: IEEE Congress on Evolutionary Computation, CEC 2006, pp. 491–498. IEEE (2006)
22. D'Haeseleer, P., Forrest, S., et al.: An immunological approach to change detection. In: Proc. of IEEE Symposium on Research in Security and Privacy, Oakland, CA (1996)
23. Majd, Mahshid, A.H., Hashemi, S.: A polymorphic convex hull scheme for negative selection algorithms. International Journal of Innovative Computing, Information and Control 8(5A), 2953–2964 (2012)
24. D'Haeseleer, P., Forrest, S., Helman, P.: An immunological approach to change detection: Algorithms, analysis and implications. In: Proceedings of the 1996 IEEE Symposium on Security and Privacy, pp. 110–119. IEEE (1996)
25. Wierzchon, S.T.: Discriminative power of the receptors activated by k-contiguous bits rule. Journal of Computer Science & Technology 1(3), 1–13 (2000)
26. de Castro, L.N., Timmis, J.: Artificial immune systems: a new computational intelligence approach. Springer (2002)
27. Ayara, M., Timmis, J., de Lemos, R., de Castro, L.N., Duncan, R.: Negative selection: How to generate detectors. In: Proceedings of the 1st International Conference on Artificial Immune Systems (ICARIS), Canterbury, UK:[sn], vol. 1, pp. 89–98 (2002)
28. González, F.A., Dasgupta, D.: Anomaly detection using real-valued negative selection. Genetic Programming and Evolvable Machines 4(4), 383–403 (2003)
29. Gonzalez, F., Dasgupta, D.: Neuro-immune and self-organizing map approaches to anomaly detection: A comparison. In: First International Conference on Artificial Immune Systems, pp. 203–211 (2002)
30. Ji, Z., Dasgupta, D.: Real-valued negative selection algorithm with variable-sized detectors. In: Deb, K., Tari, Z. (eds.) GECCO 2004. LNCS, vol. 3102, pp. 287–298. Springer, Heidelberg (2004)
31. Brownlee, J.: Artificial immune recognition system (airs)-a review and analysis. Swinburne University of Technology, Melbourne, Australia. Tech. Rep. (1-02) (2005)
32. Watkins, A., Timmis, J., Boggess, L.: Artificial immune recognition system (airs): An immune-inspired supervised learning algorithm. Genetic Programming and Evolvable Machines 5(3), 291–317 (2004)
33. Watkins, A., Timmis, J.: Artificial immune recognition system (airs): Revisions and refinements. In: AISB 2004 Convention, p. 18 (2002)
34. Watkins, A., Timmis, J.: Exploiting parallelism inherent in AIRS, an artificial immune classifier. In: Nicosia, G., Cutello, V., Bentley, P.J., Timmis, J. (eds.) ICARIS 2004. LNCS, vol. 3239, pp. 427–438. Springer, Heidelberg (2004)
35. De Castro, L.N., Von Zuben, F.J.: Learning and optimization using the clonal selection principle. IEEE Transactions on Evolutionary Computation 6(3), 239–251 (2002)
36. Brownlee, J.: Immunos-81 the misunderstood artificial immune system, ciscp, faculty of ict, swinburne university of technology. Technical report, Australia, Technical Report 1-02 (2005)

Content Based Image Retrieval Using MPEG-7 and Histogram

Muhammad Imran[1], Rathiah Hashim[1], and Noor Elaiza Abd Khalid[2]

[1] University Tun Hussein Onn Malaysia, 86400 Parit Raja, Batu Pahat Johor, Malaysia
[2] Universiti Teknologi MARA, Malysia
malikimran110@gmail.com, radhiah@uthm.edu.my, elaiza@tmsk.uitm.edu.my

Abstract. Rapid development of multimedia technologies made Content Based Image Retrieval (CBIR) an energetic research area for the researchers of multimedia domain. Texture and color features have been the primal descriptors for images in the field of CBIR. This paper proposed a new CBIR system by combining the both color and texture features. Color Layout Descriptor (CLD) from MPEG-7 is used for the color feature extraction while, Mean, variance, skewness, Kurtosis, energy and entropy are used as texture descriptors. Experiments are performed on Coral Database. The results of the proposed method namely CLD-*fos* are compared with the four well reputed systems (i.e. SIMPLIcity, Histogram based, FIRM, and Variance Segment etc) from the industry. The results of the CLD-*fos* demonstrated high accuracy rate than the previous systems during the simulations. The proposed CLD-fos achieved significant performance in terms of accuracy.

Keywords: CBIR, Image Retrieval, MPEG-7, Histogram based, First Order statistics. Entropy based CBIR. Histogram based.

1 Introduction

In this era, we have a lot of information available in the form of digital content. With every passing day, the electronic storage and the computing power is increasing. The end result of this rapid increase in content storage is making it easier for the users to access information in the form of images and videos. The image and video contents provides basis for the many educational and commercial applications. Currently, we have databases of images and videos available on a large scale but to search relevant images from them is quite challenging task. There are two major systems available to retrieve information from image databases. One is the Keyword Based Image Retrieval (KBIR) and the other is Contents Based Image Retrieval (CBIR). Most of the time, KBIR returns irrelevant images because keyword based search is highly dependent on who adds it and also the same image can have different meanings for different peoples. KBIR retrieves images manually which is a very time consuming task [1]. KBIR search the similar images using keywords while, CBIR retrieves similar images in terms of contents. Although CBIR is a better way to retrieve the relevant image

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_43, © Springer International Publishing Switzerland 2014

from any database but still there is lot of challenges to be answered in this area of research. One of the biggest challenges is to retrieve an image with high accuracy from a database.

The large collections of digital images are being created in different areas such as government, commerce, hospitals and academia. In the past, simple browsing or KBIR was used to search images from these collections. To search images with CBIR, user has to provide the image or sketch as a query to system and the system will return the similar images based on the matching features. CBIR is an application of computer vision technique which is also known as Query by Image Content (QBIC) and Content Based Visual Information Retrieval system (CBVIR).

There are different CBIR techniques proposed in the literature. Few of them, used local features while others used global features. The researchers also segment the image into regions based on color and texture to extract local features. Many machine learning techniques are also applied on the CBIR system.

In this paper we have proposed a new signature for the CBIR by using Colour Layout Descriptor (CLD) from MPEG-7 and histogram for texture features. The organization of the paper is as fallows section 2 reports the research related work, section 3 illustrate the proposed approach, section 4 describes the methodology and finally the results and conclusion are deliberated in section 5 and 6 respectively.

2 Related Work

Similarity measure is very important for any CBIR system. Beecks et al [2] performed a comparison study of the similarity measures. The authors evaluated the performance of Hausdorff Distance, Perceptually Modified Hausdorff Distance, Weighted Correlation Distance, Earth Mover's Distance, and Signature Quadratic Form Distance on four different databases available in the literature. The results of their experiments depicted that the precision rate of each similarity measure depends on the feature space and the database. In case of only color features, Perceptually Modified Hausdorff Distance (PMHD) and the EarthMover's Distance (EMD) perform well. Hausdorff Distance achieved the lowest average precision. Authors also checked the computation time for each similarity measure technique. The lowest computation time was taken by Hausdorff Distance and Perceptually Modified Hausdorff Distance whilst the lowest efficiency was shown by the EMD.

Finally, authors conclude that Quadratic Form Distance achieved the highest precision while HD and MHD reveal the lowest computation time.

Singhai and shandilya [3] performed a survey on the functionality of the CBIR. They conclude that most of the system used comprised of color and texture features while a few came up with the shape features and little bit are with layout features.

Akgül et al. [5] completed a survey of CBIR in the medical imaging. Authors discussed the current state of the art techniques of CBIR in medical imaging. They came up with the new challenges and opportunities of CBIR in medical diagnoses process.

Authors tried to focus the attention of the researcher on operation issues in medical CBIR and proposed certain strategies to tackle them. Huang et al [6] proposed the new technique of the CBIR using color moment and Gabor texture feature. To obtained the color moment they convert the RGB image to HSV image, then by getting the equalized histogram of the three HSV components calculated the three moments for each color space. Modified form of the Euclidian distance was used to measure the similarity between the query image and the database image. The equation is given below;

$$D(q,s) = \frac{1}{L}\sum_{i=0}^{L-1}\left(1 - \frac{|q_i - s_i|}{max(q_i, s_i)}\right)$$

The global distance is computed as the weighted sum of similarities as:

$$D(q,s) = \frac{\omega_c . D_{c(q,s)} + \omega_t . D_t(q,s)}{\omega_c + \omega_t}$$

Through experiments Huang et al. [6] proved that the overall result of the proposed technique was better than other techniques. Zhao and Tang [7] combined relevance feedback with SVM. The different feature combination is also used by him. They extract three different kinds of texture feature. The three texture features were combined with the color feature in different combinations. Three different combinations were contracted having one color moment and two texture features. The effect of region based filtering tested by the Pujari and Nayak [8]. Pujari used the wavelet based texture features while for the color images the texture features of each color space R G B ware extracted separately. For similarity measure the integrated region matching used. The experiments performed using 0%, 3%, 6% and 8% filtering. There is no measureable difference between the 6% and 8%. Oliveira et al [9] used breast density for the image retrieval to help the radiologists in their diagnosis. Particle swarm optimization (PSO) with relevance feedback was used by the Broilo [10] to enhance the performance of CBIR. Broilo formulates the image retrieval processes as an optimization problem and applied PSO on CBIR. PSO was proposed by Kennedy and Eberhart \cite{21-Kennedy-1995} in 1995 and modified by different research works such as [11-13].

3 The CLD-*fos* Approach

To get the comprehensive information of the image, color and texture features should be used in combination. The combining of both color and texture not only are capable to extract more image information, but also helps to describe image from the multiple aspects for more detailed information in order to obtain better search results. The proposed method is based on the CLD of MPEG-7 and first order statistics of the histogram.

3.1 MPEG-7

The MPEG-7 Visual Standards specifies content-based descriptors that allow users to measure similarity in images or video based on visual criteria, and can be used to efficiently identify, filter, or browse images or videos based on visual content [14]. MPEG-7 has different descriptor for color, texture, shape and motion. In this paper, we used the CLD and defined the signature to extract color features.

3.2 Color Layout Descriptor (CLD)

To describe the spatial distribution of color in an arbitrary shaped region CLD is the best descriptor [14]. There are four stages to extract the CLD [15]. In the first stage input image is partitioned into 64 (8x8) blocks. In second stage a single representative color is selected for each block. As a result a tiny image representation of size 8x8 is obtained. In the third stage each of the three color components are transformed by 8x8 DCT. Three sets of 64 DCT coefficients are obtained. To calculate the DCT in a 2D array, the formulas below are used.

$$B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos\frac{\pi(2m+1)_p}{2M} \cos\frac{\pi(2n+1)_q}{2N} \qquad \begin{matrix} 0 \le p \le M-1 \\ 0 \le q \le N-1 \end{matrix}$$

$$\alpha_p = \begin{cases} {}^{1}\!/\!{}_{\sqrt{M}}, p = 0 \\ \sqrt{{}^{2}\!/\!{}_{M}}, 1 \le p \le M-1 \end{cases} \qquad \alpha_q = \begin{cases} {}^{1}\!/\!{}_{\sqrt{N}}, q = 0 \\ \sqrt{{}^{2}\!/\!{}_{N}}, 1 \le q \le N-1 \end{cases}$$

The values B_{pq} are called the DCT coefficients of A. In the final stage zigzag scanning is performed for each set of coefficients.

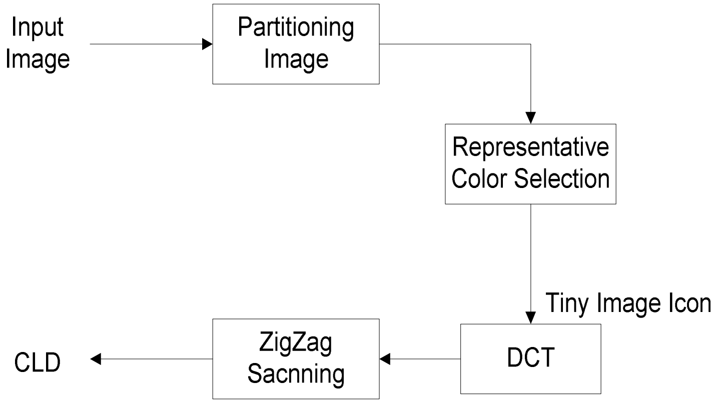


Fig. 1. Extraction process of the CLD

3.3 Color Feature Vector

After zigzag scanning we obtain three matrices for each block of Y, Cb and Cr color space. Three feature vectors for an image can be obtained by taking the sum of the each matrix. The resulting feature vector is obtained by horizontally concatenating the three feature vector.

3.4 Histogram Based Texture Feature

To extract the texture features we used the first-order histogram based features. Using original image values the first order texture measure are calculated. When an image is represented as a histogram, the intensity value concentration on all or part of the histogram is used for the texture analysis [13]. Different features can derived from histogram based approach include moments such as variance, mean, skewness, entropy, Kurtosis and Energy etc [13]. Fallowing equations are used to calculate these moments.

$$Mean: \quad \mu = \sum_{i=1}^{L} k_i \, p(k_i)$$

$$Varience: \quad \sigma^2 = \sum_{i=1}^{L} (k_i - \mu)^2 \, p(k_i)$$

$$Skewness: \quad \mu_3 = \sigma^{-3} \sum_{i=1}^{L} (k_i - \mu)^3 p(k_i)$$

$$Kurtosis: \quad \mu_4 = \sigma^{-4} \sum_{i=1}^{L} (k_i - \mu)^4 P(k_i) - 3$$

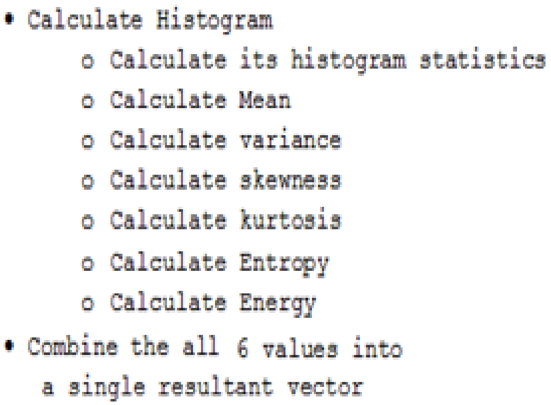
$$Energy: \quad E = \sum_{i=1}^{L} [\boldsymbol{p}(\boldsymbol{k_i})]^2$$

$$Entropy: \quad H = -\sum_{i=1}^{L} \boldsymbol{p}(\boldsymbol{k_i}) \, \boldsymbol{log_2}[\boldsymbol{p}(\boldsymbol{k_i})]$$

where k_i= gray value of the i^{th} pixel L= the number of distinct gray levels p (k_i)= normalized histogram. According to [14] average level of intensity of the image is represented by the mean. Variation of intensity around the mean is defined as variance. If the histogram is symmetrical about the mean, the skewness reaches to zero and is otherwise either positive or negative depending whether it has been skewed above or below the mean. The flatness of the histogram is measured as kurtosis; to normalize μ_4 to zero for a Gaussian-shaped histogram component 3 is inserted. Histogram uniformity is measure by the entropy. The entropy is a measure of histogram uniformity.

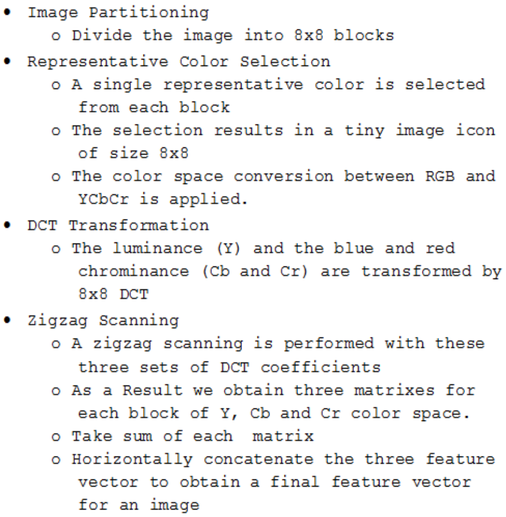
4 Methodology

4.1 Texture Feature Extraction Algorithm



```
• Calculate Histogram
      o Calculate its histogram statistics
      o Calculate Mean
      o Calculate variance
      o Calculate skewness
      o Calculate kurtosis
      o Calculate Entropy
      o Calculate Energy
• Combine the all 6 values into
   a single resultant vector
```

4.2 Color Feature Extraction Algorithm



```
• Image Partitioning
    o Divide the image into 8x8 blocks
• Representative Color Selection
    o A single representative color is selected
      from each block
    o The selection results in a tiny image icon
      of size 8x8
    o The color space conversion between RGB and
      YCbCr is applied.
• DCT Transformation
    o The luminance (Y) and the blue and red
      chrominance (Cb and Cr) are transformed by
      8x8 DCT
• Zigzag Scanning
    o A zigzag scanning is performed with these
      three sets of DCT coefficients
    o As a Result we obtain three matrixes for
      each block of Y, Cb and Cr color space.
    o Take sum of each  matrix
    o Horizontally concatenate the three feature
      vector to obtain a final feature vector
      for an image
```

By using color feature extraction algorithm we extract the color features, then extract the texture feature by using texture feature extraction algorithm and combined them in a single feature vector. We extract the features of the entire image database and save to disk. When user input the query image, the system measures the similarity between query image and the database image by using Manhattan distance. After similarity measure the system sort the result and display to user. The flow chart of CLD-*fos* is shown in Figure 2.

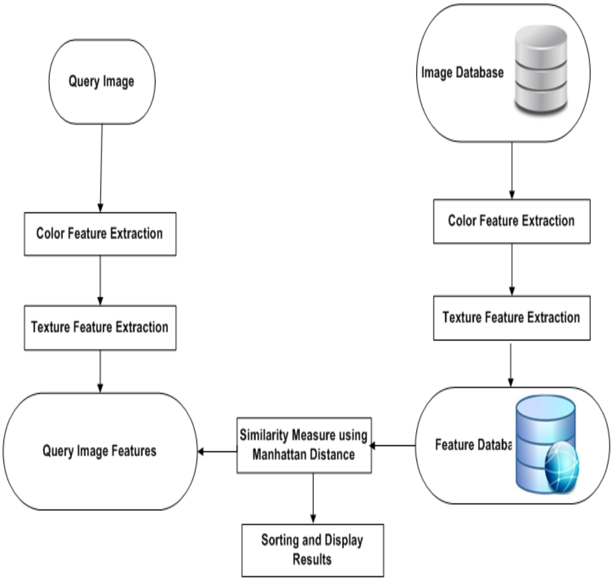

Query Image
Image Database
Color Feature Extraction
Color Feature Extraction
Texture Feature Extraction
Texture Feature Extraction
Query Image Features
Similarity Measure using Manhattan Distance
Feature Databa
Sorting and Display Results

Fig. 2. Flow chart of proposed Techniques

5 Results and Analysis

We compared the result of the proposed system with other standard benchmark systems namely SIMPLIcity, FIRM, Variance Segment Method [16-18] and Histogram based taken from FEI [18]. We used the famous Coral Database for our experiments. The database contains 10 classes and each class has 100 images. We have implemented the CLD-*fos* system under Matlab R2010b.

5.1 Metrics Used for the Evolution

To measure the performance of the CBIR system different metrics are available. Precision is one of the metric which has been used in the several previous works such as Hiremath et al [19], Banerjee et al [16] and Wang et al [17]. Precision can be calculated as;

$$Precision = \frac{Number\ of\ True\ Positive}{Number\ of\ True\ Possitive\ +\ False\ Possitive}$$

Recall is also one of the metrics used for the evolution of the CBIR system.

$$Recall = \frac{Number\ of\ Tru\ Positive}{Total\ number\ of\ True\ positve}$$

At the end in Figure 10 we have the graph representing the comparison of CLD-*fos* with variance segment method in term of recall.

5.2 Performance in Terms of Precision

As illustrated above, precision is one of the metric used to check the performance of the CBIR system. Table-1 shows the performance of the proposed algorithm with different P @ n evaluation. The precision is calculated using the top most 50, 30 and 10 ranked results. In our experiments, we repeated the search for 10 random images in each category and took the category wise average.

Table 1. Performance at different n

Class	The Precision at n (calculated using n top most results)		
	n=10	**n=30**	**n=50**
Africa	.77	.69	.65
Beach	.62	.57	.46
Buildings	.37	.3	.25
Buses	.45	.34	.35
Dinosaurs	1	1	1
Elephant	.56	.45	.36
Flower	.74	.7	.55
Horses	.69	.53	.36
Mountains	.37	.34	.26
Food	.32	.32	.29
Avg	**.589**	**.524**	**.453**

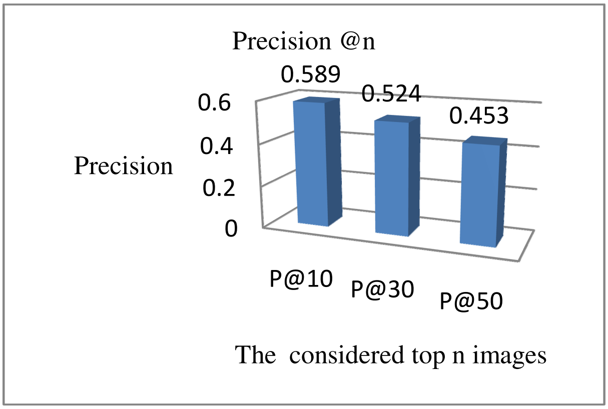


Fig. 3. Performance at different n

From Table-1 and the graph shown in Figure-3, it is clear that precision is affected, if we change the number of top most n images. However, it is observed that if we consider the n=30 then the result generated by the proposed method are better than SIMPLIcity, Simple Hist, FIRM and Variance Segmentation.

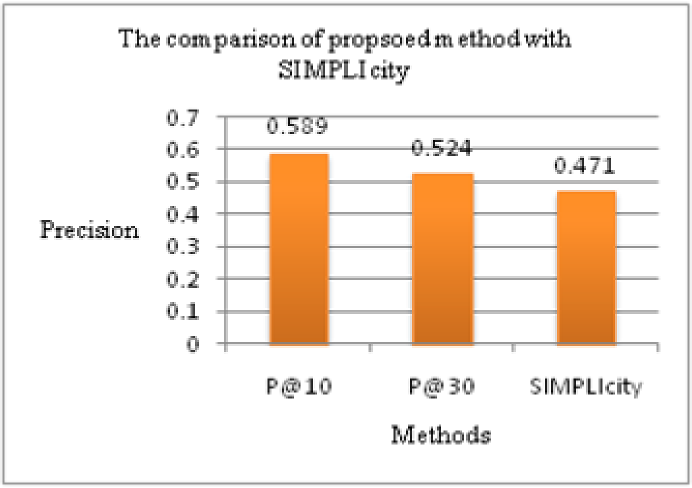


Fig. 4. Comparison with SIMPLIcity

5.3 Comparison with Previous Methods

In the table 2, we have shown the results of the proposed system with previously reported four systems. As the Coral Database has 10 classes, so we computed the average result of each class.

Table 2. Comparison of proposed method with previous methods

Class	SIMPLIcity	Simple Hist	FIRM	Variance Segment	Proposed Method
Africa	.48	.30	.47	0.13	.69
Beach	.32	.30	.35	0.26	.57
Buildings	.35	.25	.35	0.11	.3
Buses	.36	.26	.60	0.17	.34
Dinosaurs	.95	.90	.95	0.96	1
Elephant	.38	.36	.25	0.34	.45
Flower	.42	.40	.65	0.49	.7
Horses	.72	.38	.65	0.20	.53
Mountains	.35	.25	.30	0.25	.34
Food	.38	.20	.48	0.15	.32
Avg	**.471**	**.36**	**.505**	**0.174**	**.524**

The SIMPLIcity is cited in several earlier works and based on segmentation therefore have better performance. From Table-2, it is clear that SIMPLIcity has better results than other methods. But the proposed methods have best performance then all 4 methods listed in the above table including SIMPLIcity.

In the graph shown in Figure-5, the proposed method is compared with SIMPLIcity for n = 10 and 30. From the above graph, it is clear that the proposed method has better precision than SIMPLIcity. The SIMPLIcity used sophisticated set of features. But our system with simple texture and MPEG-7 descriptor outperforms.

In the Figure-5, the precision is calculated for n=30. And the average is taken for 10 images in each category and then the overall average of all categories is calculated. Here we can see clearly the result of proposed system is better than other methods.

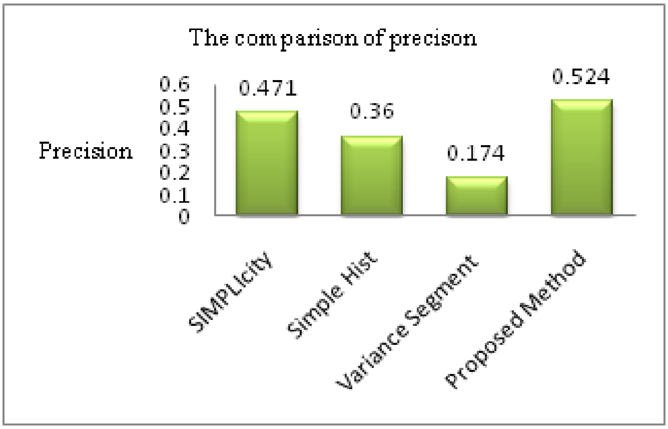


Fig. 5. The Comparison of proposed method with Earlier Methods

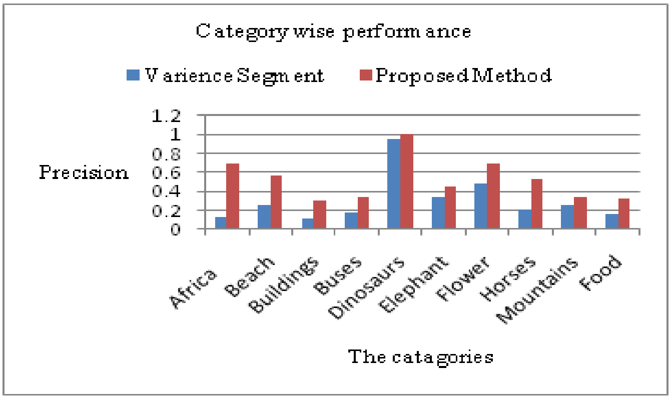


Fig. 6. Category-wise Comparison of Proposed Method (n @ 30) with Variance Segment Method

In the figure 6, we can see that the proposed method has better result than Variance Segment method on all 10 categories. For the class dinosaurs the proposed method has 100% retrieval rate as the precision reaches to 1. From graph shown in figure 7 we can see the average retrieval rate of all 10 classes using proposed system. The precision of dinosaur's class is highest as it reaches to 1, while the food class has lowest precision, except food, building and buses the proposed system has better retrieval rate for all other classes.

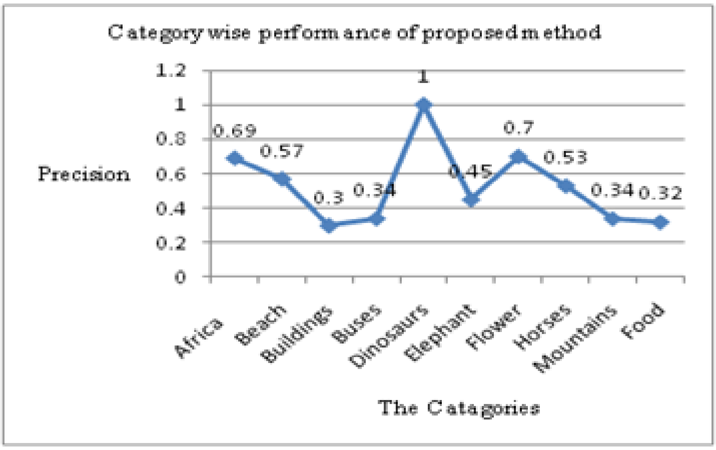


Fig. 7. Average Performance of different categories

6 Conclusion

A new CBIR system has been implemented to answer the shortcomings in the previous CBIR methods. Nowadays a lot of information in the form of digital content is easily accessible on the internet but finding the relevant image is a big problem using current CBIR systems. The proposed system has used the CLD from MPEG-7 and first order statistics. CLD was selected for the color features and texture features are extracted using first order statistics The performance of the system has been evaluated with the standard SIMPLIcity and other similar CBIR techniques. The system is also tested on the different top n image retrieval. During simulations it is observed that the proposed system outperforms, if the n is equal to 30 or less than 30. From the results, it can be clearly observed that the performance of the proposed CBIR system is better than the previous CBIR systems using sophisticated region, shape and texture matching techniques.

Acknowledgement. The researchers would like to thank University Tun Hussein Onn Malaysia (UTHM) for supporting this project under Project Vote No 1315.

References

[1] Zhi-Chun, H., et al.: Content-based image retrieval using color moment and Gabor texture feature. In: International Conference onMachine Learning and Cybernetics (ICMLC), pp. 719–724 (2010)

[2] Beecks, C., et al.: A Comparative Study of Similarity Measures For Content-Based Multimedia Retrieval. Presented at the IEEE International Conference on Multimedia and Expo (ICME) (2010)

[3] Singhai, N., Shandilya, P.S.K.: A Survey On: Content Based Image Retrieval Systems. International Journal of Computer Applications vol 4 (July 2010)

[4] Abubacker, K.A.S., Indumathi, L.K.: Attribute Associated Image Retrieval and Similarity Reranking. In: Proceedings of the International Conference on Communication and Computational Intelligence (2010)

[5] Akgül, C.B., et al.: Content-Based Image Retrieval in Radiology: Current Status and Future Directions. Journal of Digital Imaging (2010)

[6] Huang, Z.-C., et al.: Content Based Image Retrieval Using Color Moment and Gabor Texture Feature. In: Proceedings of the Ninth International Conference on Machine Learning and Cybernetics (2010)

[7] Zhao, L., Tang, J.: Content-Based Image Retrieval Using Optimal Feature Combination and Relevance Feedback. Presented at the International Conference on Computer Application and System Modeling (2010)

[8] Pujari, J.D., Nayak, A.S.: Effect of Region Filtering on The Performance of Segmentation Based CBIR System. Presented at the International Conference on Signal and Image Processing (2010)

[9] de Oliveiraa, J.E.E., et al.: MammoSys: A content-based image retrieval system using breast density patterns. Computer Methods and Programs in Biomedicine, 10 (2010)

[10] Broilo, M., Natale, F.G.B.D.: A Stochastic Approach to Image Retrieval Using Relevance Feedback and Particle SwarmOptimization. IEEE Transactions on Multimedia 12, 11 (2010)

[11] Imran, M., et al.: Modified Particle Swarm Optimization with student T mutation (STPSO). In: 2011 International Conference on Computer Networks and Information Technology (ICCNIT), pp. 283–286 (2011)

[12] Imran, M., et al.: Particle Swarm Optimization (PSO) Variants with Triangular Mutation. Journal of Engineering and Technology (2013)

[13] Imran, M., et al.: Opposition based Particle Swarm Optimization with student T mutation (OSTPSO). In: 2012 4th Conference on Data Mining and Optimization (DMO), pp. 80–85 (2012)

[14] Sikora, T.: The MPEG-7 Visual Standard for Content Description An Overview. IEEE Transactions on Circuits and Systems for Video Technology 11, 696–702 (2001)

[15] Royo, C.V.: Image-Based Query by Example Using MPEG-7 Visual Descriptors. Master, Universitat Politècnica de Catalunya (2010)

[16] Banerjee, M., et al.: Content-based image retrieval using visually significant point features. Fuzzy Sets and Systems, 3323–3341 (2009)

[17] Wang, J.Z., et al.: SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture Libraries. IEEE Transactions on Pattern Analysis And Machine Intelligence 23 (2011)

[18] Chen, Y., Wang, J.Z.: A region-based fuzzy feature matching approach to content-based image retrieval. IEEE Transactions on Image Processing Pattern Analysis and Machine Intelligence 24, 1252–1267 (2002)

[19] Hiremath, P.S., Jagadeesh Pujari, J.D.: Content Based Image Retrieval using Color Boosted Salient Points and Shape features of an image. International Journal of Image Processing 2, 10–17 (2008)

Cost-Sensitive Bayesian Network Learning Using Sampling

Eman Nashnush and Sunil Vadera

The School of Computing, Science and Engineering, Salford University, Manchester, UK
E.Nashnush1@edu.salford.ac.uk, S.Vadera@salford.ac.uk

Abstract. A significant advance in recent years has been the development of cost-sensitive decision tree learners, recognising that real world classification problems need to take account of costs of misclassification and not just focus on accuracy. The literature contains well over 50 cost-sensitive decision tree induction algorithms, each with varying performance profiles. Obtaining good Bayesian networks can be challenging and hence several algorithms have been proposed for learning their structure and parameters from data. However, most of these algorithms focus on learning Bayesian networks that aim to maximise the accuracy of classifications. Hence an obvious question that arises is whether it is possible to develop cost-sensitive Bayesian networks and whether they would perform better than cost-sensitive decision trees for minimising classification cost? This paper explores this question by developing a new Bayesian network learning algorithm based on changing the data distribution to reflect the costs of misclassification.The proposed method is explored by conducting experiments on over 20 data sets. The results show that this approach produces good results in comparison to more complex cost-sensitive decision tree algorithms.

Keywords: Cost-sensitive classification, Bayesian Learning, Decision Trees.

1 Introduction

Classification is one of the most important methods in data mining; playing an essential role in data analysis and pattern recognition, and requiring the construction of a classifier. The classifier can predict a class label for an unseen instance from a set of attributes. However, the induction of classifiers from the data sets of pre-classified instances is a central problem in machine learning [1]. Therefore, several methods and algorithms have been introduced, such as decision trees, decision graphs, Bayesian networks, neural networks, and decision rules, etc. Over the last decade, graphical models have become one of the most popular tools to structure uncertain knowledge. Furthermore, over the last few years, Bayesian networks have become very popular and have been successfully applied to create consistent probabilistic representations of uncertain knowledge in several fields [2].

Cost-insensitive learning algorithms focus only on accuracy (class label output), and do not take misclassification costs or test costs into consideration. However, the

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_44, © Springer International Publishing Switzerland 2014

performance of any learning algorithm, in practice, normally has to take the cost of misclassification into account. Hence, in recent years, a significant level of attention has been paid to cost-sensitive learning, including making accuracy-based learners cost-sensitive [3, 4]. Zadrozny *et al.* [6] divide cost-sensitive classifiers into two categories: the amending approach (changing the classifier to a transparent box) and the sampling approach (using the classifier as a black box). Among all the available cost-sensitive learning algorithms, most of the work has focused on decision tree learning, with very few studies considering the use of Bayesian networks for cost-sensitive classification.

Therefore, this paper aims to explore the use of Bayesian networks (BNs) for cost-sensitive classification. During this paper, a new method known as the Cost-Sensitive Bayesian Network (CS-BN) algorithm, which uses a sampling approach to induce cost-sensitive Bayesian networks, is developed and compared with other, more common approaches such as cost-sensitive decision trees. This paper is organized as follows: in section 2 we will provide a number of definitions and background information on cost-sensitive learning algorithms. Section 3 will introduce some of the previous work on the sampling approach. In section 4, we will present our method for converting the existing BN algorithm into a CS-BN by changing the number of examples to reflect the costs. In section 5 we present the results obtained by carrying out an empirical evaluation on data from the UCI repository. Finally, section 6 will provide a conclusion, along with a summary of the main contribution of this paper.

2 Cost-Sensitive Learning Perspective and Overview

A good cost-sensitive classifier should be able to predict the class of an example that leads to the lowest expected cost, where the expectation is computed after applying the classifier by using the expected cost function, as shown in the following equation [6,21]:

$$expected\ cost(x|i) = \sum_j C(i,j)P(j|x). \qquad (1)$$

Where *P(j|x)* represents the probability of an example being in class j given it is actually of class *x*, and *C(i,j)* represents the cost of misclassifying an example as class *i* when it is in class *j* [21]. In particular, cost-sensitive algorithms aim to minimize the number of high-cost misclassification errors, thus reducing the total number of misclassification errors. According to Zadrozny*et al.* [6], cost-sensitive classifiers can be divided into two categories: *Black Box* (sampling), and *Transparent Box.* Black box methods use a classifier as a black box, and use resampling methods according to a class weight. On the other hand, transparent box methods use weights to change the classifier learning algorithm directly. Conversely, Sheng and Ling [7] used different terms such as *direct method*, and *wrapper methods,* as shown in Figure 1.

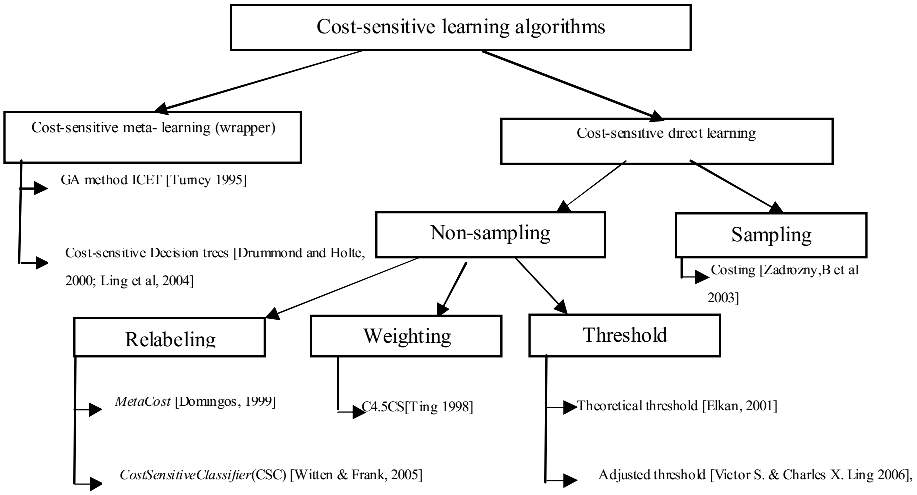

Cost-sensitive learning algorithms
Cost-sensitive meta- learning (wrapper)
GA method ICET [Turney 1995]
Cost-sensitive Decision trees [Drummond and Holte, 2000; Ling et al, 2004]
Cost-sensitive direct learning
Non-sampling
Sampling
Costing [Zadrozny,B et al 2003]
Relabeling
Weighting
Threshold
MetaCost [Domingos, 1999]
CostSensitiveClassifier(CSC) [Witten & Frank, 2005]
C4.5CS[Ting 1998]
Theoretical threshold [Elkan, 2001]
Adjusted threshold [Victor S. & Charles X. Ling 2006],

Fig. 1. Cost-sensitive learning category (Shendand Ling [7])

AsZadrozny [6] points out, wrapper methods (Black Box), deal with a classifier as a closed box, without changing the classifier behaviour, and can work for any classifier. In contrast, direct methods (Transparent Box), require knowledge of the particular learning algorithm, and can also work on the classifier itself by changing its structure to include the costs.

3 Review of Previous Work on Sampling Approach

Most studies regarding cost-sensitive learning have used direct methods or sampling methods, and most have focused on decision tree learning. This section briefly reviews some of these methods. In addition, this section describes different methods of cost-sensitive learning by changing the data distribution to involve costs and solve an unbalanced data distribution problem, where, for example, the number of negative examples is significantly less than the number of positive examples. Several literature reviews show different methods, where some of them amend the number of negative examples (*over-sampling*); some of them change the number of positive examples (*under-sampling*); a few of them use the "*SMOTE*" (Synthetic Minority Over-Sampling Technique) algorithm that tackles the imbalanced problem by generating synthetic minority class examples [8]; and others use a "*Folk Theorem*" [5, 21]that amends the distribution according to the cost of misclassification.

Kubat and Matwin [12] used one side selection by under-sampling the majority class while keeping the original population of the minority class. As Elkan [21] pointed out in 2001, changing the balance of negative and positive training examples will affect classification algorithms. Ling and Li [13] combined over-sampling with under-sampling to measure the performance of a classifier. Domingos [14] introduced the MetaCost algorithm which is based on sampling with labeling and bagging. MetaCost uses the resampling with replacement to create a different sample size, then estimates each example in the same sample size by voting each example in different samples, where the number of instances in each resamples is smaller than the training size, and then applies an equation (1) to re-label each training example with the optimal class estimation. Finally, it reapplies the classifier again, on the new relabelled training data set [15]. Figure 2 summarises the MetaCost algorithm [15].Domingos concluded that this algorithm provides goods resultson large data sets. In addition, most researchers have dealt with this problem by changing the data distributions to reflect the costs, though most of them utilize a decision tree learner as a base learner, and the reader is referred Lomax and Vadera [4] for a comprehensive survey of cost sensitive decision tree algorithms for details.

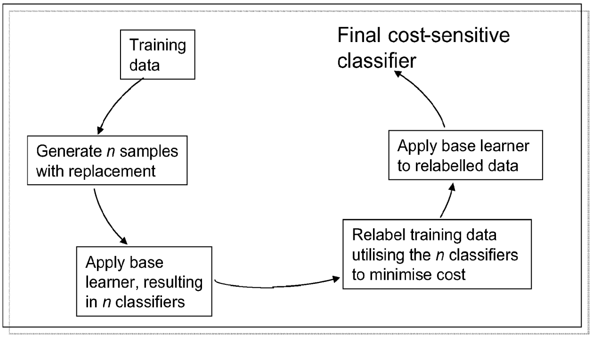

Training data
Generate *n* samples with replacement
Apply base learner, resulting in *n* classifiers
Relabel training data utilising the *n* classifiers to minimise cost
Apply base learner to relabelled data
Final cost-sensitive classifier

Fig. 2. The MetaCost system [15]

4 New Cost-Sensitive Bayesian Network Learning Algorithm via the Distributed Sampling Approach

A survey of the literature shows that, to date, there are very few publications regarding cost sensitive Bayesian networks (CS-BNs), but plenty on cost-sensitive decision tree learning. This section presents a sampling approach used to develop CS-BNs and presents the use of distributed sampling to take account of misclassification costs and reduce the number of errors. Thus, the compelling question, given the different class distributions, is: what is the correct distribution for a learning algorithm?

In response, it has been observed that naturally-occurring distributions are not always the optimal distribution [8]. In our experiments, we used the sampling (Black Box) method, because this method can also be used to address the imbalanced data problem and can be applied to any learning algorithm. In our study, we used Folk Theorem to change the data distribution. This approach has previously been introduced by Zadroznyet *al.* [5]. This theorem draws a new distribution from the old distribution, according to cost proportions, to change the data distribution and obtain optimal cost-minimization from the original distribution. This theorem is only theoretically motivated, and does not require any probability density estimation. Thus, we have used this theorem on the BN classifier, which has not been used before in this classifier.

4.1 Use of the Folk Theorem for CS-BNs

This method can be applied to any cost-insensitive classification algorithm to form a cost-sensitive classification algorithm. This method can be conducted by reweighing the instances from the training example and then using that weight on the classification algorithm. The Folk Theorem is used to change the data distribution to reflect the costs. Zadrozny*et al.*, [5] stated that "*if the new examples are drawn from the old distribution, then optimal error rate classifiers for the new distributions are optimal cost minimizes for data drawn from original distribution.*" This is shown in the following equation (2) [5]:

$$D'(\mathrm{x},\mathrm{y},\mathrm{c}) = \frac{\mathrm{C}}{\mathrm{E}_{\mathrm{x},\mathrm{y},\mathrm{c}\sim \mathrm{D}[\mathrm{c}]}} \mathrm{D}(\mathrm{x},\mathrm{y},\mathrm{c}). \tag{2}$$

Where, the new distribution D' = factor * Old distribution D; x is instance; y is the class label; and C is the cost according to misclassified instance x. Technically, the optimal error rate classifiers from D' are the optimal cost minimizers from the data, which have been drawn from D. This theorem creates new distribution from the old distribution by multiplying old distribution with a factor proportional to the relative cost of each example, and the new distribution will be adapted with that cost. Therefore, this method enables the classifier to obtain the expected cost minimization from the original distribution and, in the worst case scenario; this method can be guaranteed a classifier to provide a good approximate cost minimization for any new sample.

However, there are different types of BNs, as well as methods for learning them. Given their efficiency compared to full networks, we used a search algorithm to construct*Tree Augmented Naive Bayes Networks* (TANs), along with *Minimum Description Length* (MDL), which was introduced by Fayyad and Irani [19] to calculate the score of information between links in a tree.

In our experiments, we attempted to change the proportions of instances (samples) in each class label, according to its cost, by using the above Folk Theorem [5]. In the current experiment, we used a constant cost of 1:4, where we assigned the common majority class cost to 1 and other, minority class cost to 4. The following steps were conducted with the CS-BN by using a sampling approach:

- **Splitting:** Data are split into a training set and testing set. The training set uses 75% of the original data, while the testing set uses 25% of the original data.
- **Cost proportion:** According to cost proportions, the new data distribution should be calculated as being equal to these proportions. For instance, if the cost of wrongly classifying a sick patient as healthy is £20 and the cost of misclassifying a healthy patient as sick is £2, then the cost proportion of the sick class will be 20/22=0.90. In our experiments we used cost proportion by assigning rare class cost to 4 and common class cost to 1. Thus, the cost proportion in our algorithm would be 0.8 and 0.2 respectively, based on equation (3):

$$CostProportionofclass_i = \frac{Cost_i}{\sum_{j=0}^{k} Cost_j}. \tag{3}$$

Where i and j is the class index and k is the number of classes.

- **Changing Proportion:** This involves changing the training data distributions according to the cost ratio of each class. For example, when the costs are 1:4, the new proportions on the training set for each class will be 20% and 80% respectively.

There are different methods that can be used to achieve sampling. During our research, we used two methods, as discussed in section 3 of this paper. These methods were *under-sampling* and *over-sampling*. Obviously, where the new proportion was less than the original proportion, we used under-sampling (without replacement) to delete some of the examples in the frequent class. On the other hand, if the new proportion was greater than the original proportion, we used over-sampling (with replacement) by making a random generation of new instances which belonged to the rare class, and increasing the number of examples. As a result, the training data required further resampling according to their costs. Finally, we used the original BN classifier on the training data, followed by using the testing set with the original distribution (without changing any instances) to evaluate the training model.

However, Figure 3 presents the pseudo-code of our method (i.e. CS-BN with sampling approach):

```
 CS-BN via sampling approach:
1. Divide dataset into 75% of instances for training, and 25%
   for testing. With the same class distributions.
2. Changing the data distribution according to the cost
   proportions in each class,  $CostProportion = \frac{costi}{\sum_{i,j} costi,j}$
3. Using Bayesian network algorithm (TAN to learn structure).
4. Evaluating the model on the original test set
   distribution.
```

Fig. 3. Cost-Sensitive Bayesian Network Algorithm by sampling

4.2 Experiment

Our experiment demonstrates how changing the distribution of data will affect the performance and cost of a Bayesian classifier. We experimented with 24 data sets from the UCI repository [20]. To evaluate the performance of our proposed method, we used the original testing distribution. An evaluation was carried out in order to compare CS-BN with existing algorithms implemented in WEKA:(i) Original Bayes Net (that implemented by TAN) (Friedman *et al.* [1], Version 8); (ii) Decision Tree Algorithm J48 (which is their implementation of C4.5, Version 8); and (iii) MetaCost with J48 as the base classifier [14].

Table 1 presents the results of the CS-BN algorithm via changing distributions (Black Box), and the original BN algorithm. It also shows the comparison between the original Bayes Net (TAN), existing algorithm (decision tree J48), and MetaCost with j48 classifier. The proposed algorithm produced lower costs for cost matrix 1 and 4 on most of the data set. In our experiment, we noticed that number of False Negative (rare) with our Black Box method was less than number of False Negative (rare) of the existing BN algorithm; thus, the total cost will be reduced to approximately 933 units.

Table 1. Comparison between CS-BN via changing the distributions and existing algorithms

	Bayes Network after change proportions		**Original Bayes Network**		**Metacost j48**		**Original Decision tree J4.8**	
Dataset	accuracy	Cost	Accuracy	Cost	Accuracy	Cost	accuracy	cost
Gymexamg	40.051	**442**	69.11	758	41.560	490	65.28	710
Spambase	91.506	**205**	93.26	212	91.768	241	92.03	238
German	59.126	**118**	71.43	201	70.634	143	71.82	218
tic-tac	54.958	118	73.97	171	81.818	**89**	84.29	92
Breast	97.752	**4**	97.75	7	94.943	21	94.94	21
Breastcance	45.833	**48**	68.05	68	27.777	58	63.88	77
bupa_liver	42.045	**51**	57.95	148	56.818	56	63.63	71
Crx	82.080	**49**	86.71	62	83.236	47	87.28	61
Diabetes	60.621	**94**	77.72	100	73.056	118	74.09	134
Heart	79.710	**20**	84.06	29	78.260	42	78.26	42
Hepatise	73.170	11	92.68	**6**	80.487	11	70.73	27
horse-colic	64.705	**27**	72.06	76	70.588	32	66.17	71
Horse	71.27	57	82.98	55	77.659	60	79.78	64
Hypo	98.108	**18**	98.87	18	98.865	27	98.99	26
Iono	87.777	**26**	88.89	31	86.666	30	86.66	30
Ionosphere	87.777	**29**	90	33	81.111	38	81.11	38
Labor	73.333	**10**	93.34	10	80	9	80	9
Mushroom	99.787	12	99.79	12	99.787	12	99.78	12
Pima	65.284	**91**	76.69	123	72.538	101	72.53	101
Sonar	62.962	**38**	62.96	65	61.111	63	61.11	63
Unbalanced	98.130	13	98.59	12	98.598	12	98.59	12
Weather	40	3	40	3	60	**2**	60	2

5 Results and Discussion

This experiment shows that the number of misclassifications of rare class (more expensive) are always less than the number of misclassifications for the rare class in the original TAN algorithm for most of the data. Thus, the results are alwaysbetter in terms of cost, as we can see in Table 1. Furthermore, as shown in Figure 4, for most of the data sets, the changing proportion method (CS-BN via sampling) gives good results compared to the original TAN, MetaCost, and Decision Tree (j48). On the other hand, in Figure 5, it is shown that the accuracy, in most cases, is a little lower than the original TAN algorithm.

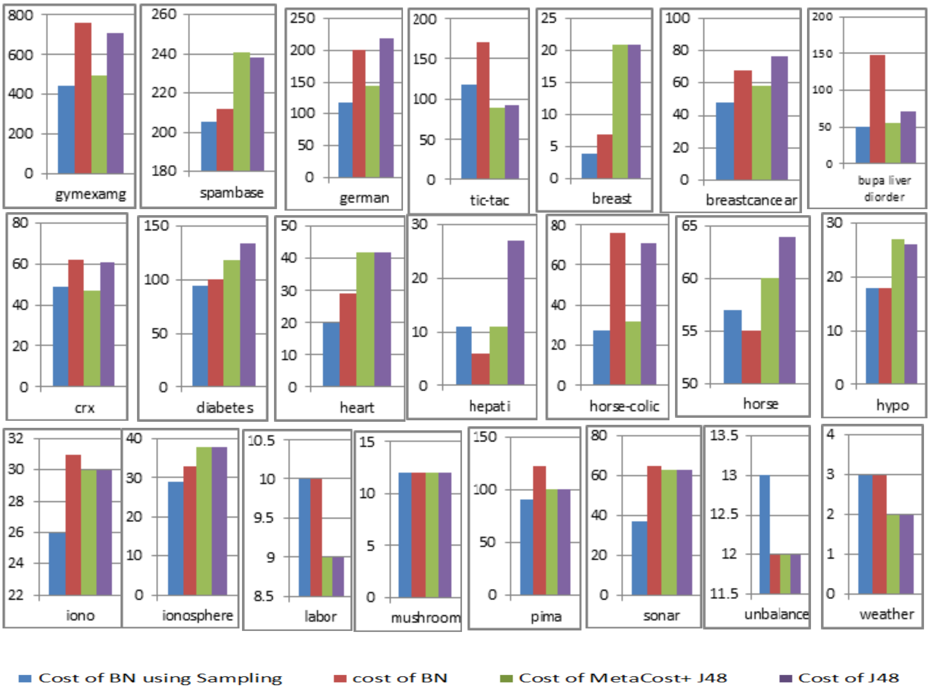


Fig. 4. Expected cost of CS-BN via changing the distributions and existing algorithms

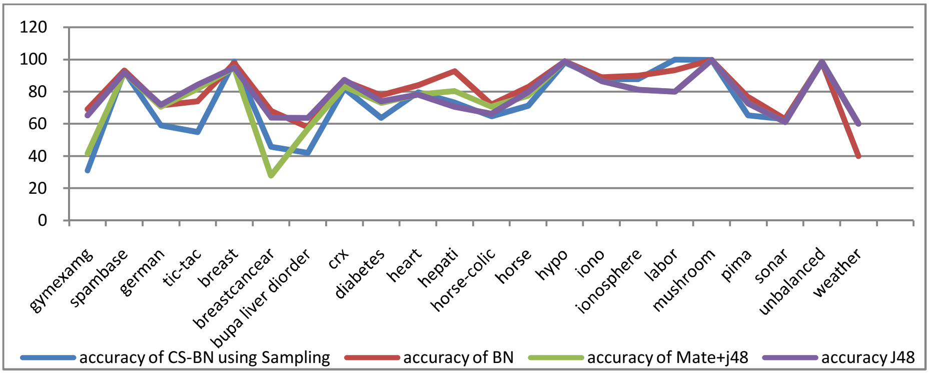


Fig. 5. Accuracy of CS-BN via changing the distributions and existing algorithms

As consequence, changing the data distributions before applying TAN classifier yields good results in most data; especially if the data are not very highly skewed to one class. Therefore, the expected cost of using our experiments will provide a reduction of misclassification costs, compared to the original algorithm, which does not use this method. Therefore, we believe that theproposed CS-BN approach of changing the data distributions will produce good results in terms of cost and accuracy.

6 Conclusion

Although much work has been conducted on the development of cost-sensitive decision tree learning, little has been conducted on assessing whether other classifiers, such as Bayesian networks, can lead to better results. Therefore, taking into account work with the folk theorem [22,5], a new Black Box method, based on amending the distribution of examples to reflect the costs of misclassification, was applied in order to develop cost-sensitive Bayesian networks. A preliminary experiment, amending the distributions of TAN, has been carried out on several datasets previously studied by various researchers using different methods.

Our *CS-BN withsampling*approach has been evaluated and compared with MetaCost+J4.8, standard decision tree (J48), and standard Bayesian networks approaches. The results for over 20 data sets show that the use of sampling yields better results than the current leading approach; namely, the use of MetaCost+J4.8.

In conclusion, our new CS-BN algorithm has been developed and explored by using a Black Box approach with sampling that amends the data distribution to take account of costs shows promising results in comparison to existing cost-sensitive tree induction algorithms.

References

1. Friedman, J.H.: Data Mining and Statistics: What's the connection? Computing Science and Statistics 29(1), 3–9 (1998)
2. Pearl, J.: Embracing Causality in Formal Reasoning. In: AAAI, pp. 369–373 (1987)
3. Vadera, S., Ventura, D.: A Comparison of Cost-Sensitive Decision Tree Learning Algorithms. In: Second European Conference in Intelligent Management Systems in Operations, July 3-4, pp. 79–86. University of Salford, Operational Research Society, Birmingham (2001)
4. Lomax, S., Vadera, S.: A survey of cost-sensitive decision tree induction algorithms. ACM Computing Surveys (CSUR) 45(2), 16:1–16:35 (2013)
5. Zadrozny, B., Langford, J., Abe, N.: Cost-sensitive learning by cost-proportionate example weighting. In: Third IEEE International Conference on Data Mining, ICDM 2003, pp. 435–442. IEEE (2003a)
6. Zadrozny, B., Langford, J., Abe, N.: A simple method for cost-sensitive learning. IBM Technical Report RC22666 (2003b)
7. Sheng, V.S., Ling, C.X.: Roulette sampling for cost-sensitive learning. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenič, D., Skowron, A. (eds.) ECML 2007. LNCS (LNAI), vol. 4701, pp. 724–731. Springer, Heidelberg (2007)
8. Chawla, N.V., Bowyer, K.W., Hall, L.O., Philip Kegelmeyer, W.: SMOTE: Synthetic minority over-sampling technique, pp. 1106–1813 (2011)
9. Ma, G.-Z., Song, E., Hung, C.-C., Su, L., Huang, D.-S.: Multiple costs based decision making with back-propagation neural networks. Decision Support Systems 52(3), 657–663 (2012)
10. Maloof, M.A.: Learning when data sets are imbalanced and when costs are unequal and unknown. In: ICML-2003 Workshop on Learning from Imbalanced Data Sets II, vol. 2, pp. 2–1 (2003)

11. Drummond, C., Holte, R.C.: C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In: Workshop on Learning from Imbalanced Datasets II, p. 11 (2003)
12. Kubat, M., Matwin, S.: Addressing the curse of imbalanced training sets: one-sided selection. In: Proceedings of the Fourteenth International Conference on Machine Learning, ICML, vol. 97, pp. 179–186 (1997)
13. Ling, C.X., Li, C.: Data Mining for Direct Marketing: Problems and Solutions. In: KDD, vol. 98, pp. 73–79 (1998)
14. Domingos, P.: Metacost: A general method for making classifiers cost-sensitive. In: Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 155–164. ACM (1999)
15. Vadera, S.: CSNL: A cost-sensitive non-linear decision tree algorithm. ACM Transactions on Knowledge Discovery from Data (TKDD) 4(2), 6 (2010)
16. Pazzani, M.J., Merz, C.J., Murphy, P.M., Ali, K., Hume, T., Brunk, C.: Reducing Misclassification Costs. In: ICML, vol. 94, pp. 217–225 (1994)
17. Batista, G.E.A.P.A., Prati, R.C., Monard, M.C.: A study of the behavior of several methods for balancing machine learning training data. ACM Sigkdd Explorations Newsletter 6(1), 20–29 (2004)
18. Agarwal, A.: Selective sampling algorithms for cost-sensitive multiclass prediction. In: Proceedings of the 30th International Conference on Machine Learning, pp. 1220–1228 (2013)
19. Fayyad, U., Irani, K.: Multi-Interval Discretization of Continuous-Valued Attributes for Classification Learning. In: Proceedings of the International Joint Conference on Uncertainty in AI, pp. 1022–1027 (1993)
20. Asuncion, A., Newman, D.: UCI machine learning repository (2007), `http://archive.ics.uci.edu/ml/`
21. Elkan, C.: The foundations of cost-sensitive learning. In: International Joint Conference on Artificial Intelligence, vol. 17(1), pp. 973–978. Lawrence Erlbaum Associates Ltd. (2001)

Data Treatment Effects on Classification Accuracies of Bipedal Running and Walking Motions

Wei Ping Loh and Choo Wooi H'ng

School of Mechanical Engineering, Universiti Sains Malaysia,
14300 Nibong Tebal, Seberang Perai Selatan, Penang, Malaysia
meloh@usm.my,
coooowooi@gmail.com

Abstract. Many real-world data can be irrelevant, redundant, inconsistent, noisy or incomplete. To extract qualitative data for classification analysis, efficient data preprocessing techniques such as data transformation, data compression, feature extraction and imputation are required. This study investigates three data treatment approaches: randomization; attribute elimination and missing values imputation on bipedal motion data. The effects of data treatment were examined on classification accuracies to retrieve informative attributes. The analysis is performed on bipedal running and walking motions concerning the human and ostrich obtained from public available domain and a real case study. The classification accuracies were tested on seven classifier categories aided by the WEKA tool. The findings show enhancements in classification accuracies for treated dataset in bipedal run and walk with respective enhancements of 3.21% and 2.29% in treated data compared to the original. The findings support the integration of data randomization and selective attribute elimination treatment for better effects in classification analysis.

Keywords: Data treatment, classification accuracy, bipedal run, walk, motion, attributes.

1 Introduction

Bipedal motion analysis is related to the understanding of how bipedal animal or human moves including factors that limits their capability to walk or run in their daily activities. This analysis develops very early research interests such as in the mechanisms of bipedal running [1-2]. For instance the study of full 3-D joint kinematic data was explored in the ostrich runs [3]. Other works studied the bipedal mechanisms of an ostrich [3-4]. 3D motion analysis was applied to determine the kinematics and kinetics of the feet segments in the normal human walk [5]. Present studies also explored body joints recognition from image sequences in series of human activities [6-8]. In recent years, motion data classification analysis has been proposed as a viable tool for grouping different movement postulations [9-12]. From

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_45, © Springer International Publishing Switzerland 2014

the classified groups, data patterns, rules and mathematical formulations were further explored.

While motion data analysis can be classified, it is rather competitive to achieve good classification accuracy. The performances of motion data classification relies very much on the nature of study data. A good classification often requires qualitative data inputs. Undoubtedly, in the context of retrieving the good qualities of data, the major obstacles are having missing values, redundant attributes and noise which often require further treatment efforts.

Segmentation and feature selection at preprocessing level to improve the performances of video data classification were proposed in recent years [13-16]. Capodifferro et al. [13] introduced the concept of video segmentation and feature extraction. However, as mentioned in [13], the weakness of segmentation is that more noise will be produced. Fuzzy inference was used to preprocess video traffic data in [15]. On the other hand, Chan et al. [16] proposed a combination of data elimination and interpolation to improve the existing techniques of preprocessing approaches. Nevertheless, to date, few analyses have reported the effects of different data treatment to enhance the classification accuracies particularly for bipedal motion data.

This paper attempts to explore the data treatment approaches effects by considering the data orders, selecting appropriate attribute elimination and examining the effects of missing values. The purpose is to enhance classification accuracy as well as to determine the key factors with significant impact on data groupings. The data treatment analysis is demonstrated on two case studies: (i) public available video concerning human and ostrich run, (ii) real experimented data involving the human walk.

2 Methodology

Data treatment is used to preprocess the raw data into a reasonable and qualitative characteristics for better classification analysis. The two cases study motions consist of 18 attributes and 60 instances in which the qualities of data observed could be improved by treating the instances and attributes. For this reason, we consider three potential treatments to capture the main transient information. The first treatment considers the random orders of data classes. The second treatment is on attribute elimination to featurize the attributes. Meanwhile the third treatment imputes incomplete data instances. The characteristics of the study cases, data transformation and the classification analyses on treated and untreated data will be detailed in the following subsections.

2.1 Dataset

Case Study 1: Running Motion

The raw bipedal motion data was retrieved from public available video domain [17]. The study data concerns two experimented bipedal running motions; between an athlete with an ostrich under different conditions. In the first experiment, the athlete and the ostrich were separated by a wired fence to run on different racetracks for about 40 yards. In second experiment, both subjects were placed in the same racetrack.

Case Study 2: Walking Motion

This case study considers on spot video recordings of human walk at the roadside of a residential area. This video involved 3 females and a male as the study subjects contributing to 4 data classes. The recorded video, captured in 73 seconds, was afterwards played back to be translated into informative data.

2.2 Video Data Transformation

The study data was transformed from the video shots into series of images snapshots.

Case study 1: 30 image snapshots of 0.5s to 1.0s intervals by which distinguishable motion postures observed were recorded. These images were converted into numeric data measures based on seven body joints coordinate positions: head, elbow (L/R), knee (L/R) and foot (L/R) (Fig. 1). The coordinates of the body segments were measured using a standardized equal size square grids of 40 pixels (rows x columns).

Case study 2: 41 snapshots of distinguishable motion postures in 1.0s intervals were collected. Similar to case study 1, the images were translated into numeric data measures based on the similar seven body joints coordinate positioning measured on standardized equal size square grids of 35 pixels (Fig. 2).

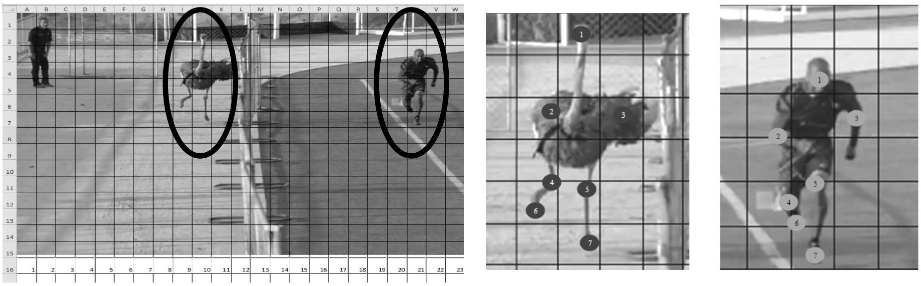


Fig. 1. Sample body segment markers on square grids of the study subjects (in circle) in case study 1

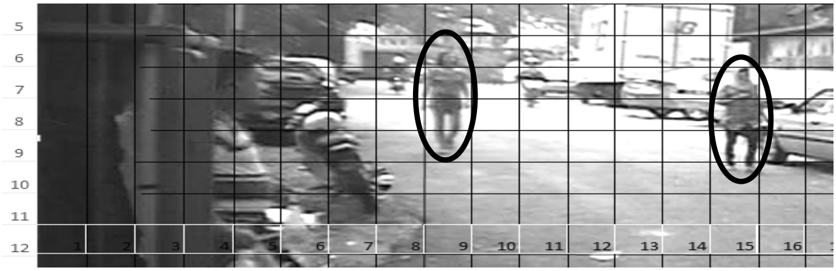


Fig. 2. Snapshot of study subjects positions (in circle) on square grids in case study 2

In case study 1, the layout of numeric tabulated data consists of 18 attributes of 60 instances: 30 rows from ‘human’ class and 30 rows from ‘ostrich’ class. Meanwhile for case study 2, the study data consists of 16 attributes of 50 instances: 9 of ‘Female1’, 9 of ‘Female2’, 19 of ‘Female3’ and 13 for ‘Male’ class.

For case study 1, the data attributes include the time, distance, speed and data class as well as the 2-D coordinates of head, elbow (L/R), knee (L/R), foot (L/R) positions. Data attributes for case study 2 are similar to that of case study 1 except for the ‘distance’ and the ‘speed’ attributes which were not considered. All attributes were recorded in ‘Numerical’ scales except for the Class attribute which is ‘Nominal’. Data layout of both study cases is shown in Table 1.

Table 1. Sample layout of raw tabulated data for cases study 1 and 2

Case study	Time (s)	HX-axis	HY-axis	LEX-axis	…	Speed (m/s)	Class
1	0.0	19.0	-1.0	17.5	…	?	Human
	⋮	⋮	⋮	⋮		⋮	⋮
	2.5	17.0	0.5	16.5		6.9438	Human
	⋮	⋮	⋮	⋮		⋮	⋮
	10.0	12.5	-4.0	8.0		?	Ostrich
2	1.0	6.5	5.5	6.0	…		Female 1
	⋮	⋮	⋮	⋮			⋮
	19.0	22.0	10.0	21.0		-	Female 4
	⋮	⋮	⋮	⋮			⋮
	13.0	10.0	7.5	8.5			Male

‘?’ - missing value.

2.3 Classification Analysis

Classification analysis was performed on seven categories of classifiers: Bayes, Function, Lazy, Meta, Misc, Rules and Trees using 66 built-in classifier algorithms for case study 1 and 62 algorithms for case study 2, supported by the Waikato Environment for Knowledge Analysis (WEKA) tool. The classification analysis yields the actual and predicted classes of data instances, correctly classified instances and misclassified instances as well as the percentages classification accuracy.

2.4 Dataset Treatment

Several dataset treatment approaches were analyzed. The approaches involved data randomization, attribute elimination and imputation of missing values (Table 2). The treated datasets characterized by these approaches were later reclassified.

Table 2. Data treatment approaches

Treatment	Description
Randomization	The orders of original data classes' columns were randomly rearranged. For instance in case study 1, the first thirty rows instances for 'human' and followed by another thirty rows of instances for 'ostrich' classes were reordered at random.
Attribute elimination	A single parameter was initially discarded (one parameter at a time) from the numeric data. Subsequently, the missing values instances were eliminated. The potential eliminations include • coordinates The entire coordinate attributes (head, elbow, knee, and foot). • H axis The 2 head segment attributes (H X-axis and H Y-axis). • E axis The 4 elbow segment attributes (LE X-axis, LE Y-axis, RE X-axis and RE Y-axis). • K axis The 4 knee segment attributes (LK X-axis, LK Y-axis, RK X-axis and RK Y-axis). • F axis The 4 knee foot attributes (LF X-axis, LF Y-axis, RF X-axis and RF Y-axis). • Distance (for Case 1) Distance attribute was removed leaving behind the Time, H X-axis, H Y-axis, LE X-axis, LE Y-axis, RE X-axis, RE Y-axis, LK X-axis, LK Y-axis, RK X-axis, RK Y-axis, LF X-axis, LF Y-axis, RF X-axis, RF Y-axis, Speed and Class attributes. • Speed (for Case 1) Speed attribute was removed leaving behind the Time, H X-axis, H Y-axis, LE X-axis, LE Y-axis, RE X-axis, RE Y-axis, LK X-axis, LK Y-axis, RK X-axis, RK Y-axis, LF X-axis, LF Y-axis, RF X-axis, RF Y-axis, Distance and Class attributes. • Time • HnE coordinate The 6 upper body segments attributes: H X-axis, H Y-axis, LE X-axis, LE Y-axis, RE X-axis, RE Y-axis. • starting time The starting time of each instance. • missing value instances All the rows with missing values in the dataset.
Imputation	All the missing values identified in the tabulated dataset were replaced with predicted values from the observed images on square grids.

3 Results and Discussion

The effects of treated and untreated data on classification performances are shown in subsections 3.1 an 3.2. Table 3 presents the average classification accuracies using seven classifiers on untreated data. Meanwhile Fig.3 shows the classification accuracies on individual data treatments by counting the number of classifiers employed. Substantial improvements in classification performances on the treated over the untreated data are shown in Fig. 4. Detail observations are discussed in the following subsections.

3.1 Original Data Classification

The variations of classification performances on original untreated data mode are summarized in Table 3.

On average, our findings show that 60 classifiers algorithms of 5 categories (Function, Lazy, Meta, Rules, Trees) and 36 algorithms of 6 categories (Bayes, Function, Lazy, Misc, Rules, Trees) indicate above 80 % correctly classified instances for case study 1 and 2 respectively. The robust accuracy obtained might be attributed to the data classes arranged in orders.

The lowest average accuracy observed was 67.5% and 76.8% on for case 1 and case 2 respectively. The misclassifications effects might be the consequence of the existing missing values and irrelevant attributes phenomenon. The existence of unimportant attributes could severely flaw the capabilities of classifier algorithms to distinguish between the motion characters of study subjects.

Table 3. Average classification accuracy by classifier category for cases study 1 and 2

Category	Case Study 1		Case Study 2	
	Number of algorithms	Average accuracy (%)	Number of algorithms	Average accuracy (%)
Bayes	4	73.3	4	97.5
Function	7	84.8	5	97.2
Lazy	4	92.1	4	91.0
Meta	27	82.4	26	76.8
Misc	2	67.5	2	97.0
Rules	9	83.9	9	85.1
Trees	13	95.8	12	95.5

3.2 Treated Data Classification

Fig. 3 presents graphs of number of classifier count (out of 66 and 62 algorithms in respective of case study 1 and case study 2) versus the types of data treatment imposed.

The different data treatments imposed observed three major changes in classification accuracy behaviours whether the classification accuracies remain, decrease or increase from the original untreated data classification results. (Fig.3).

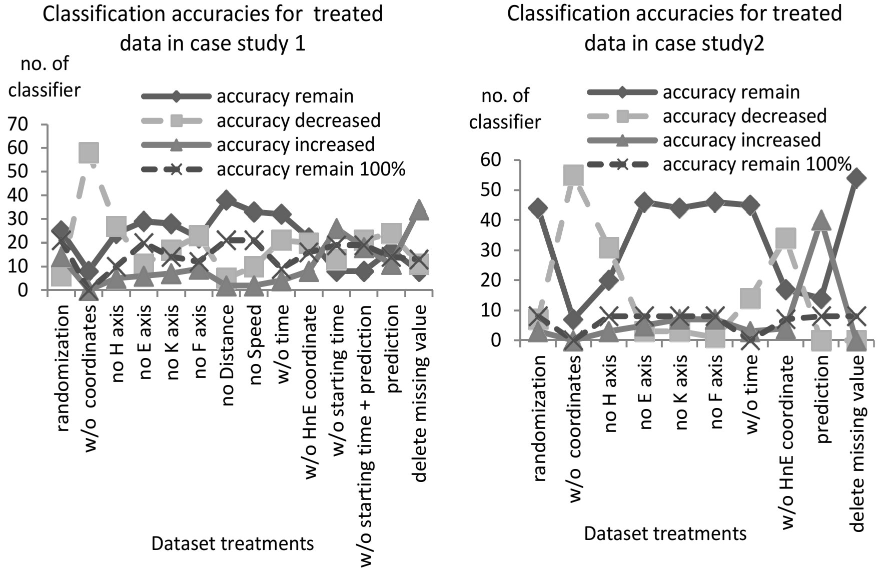
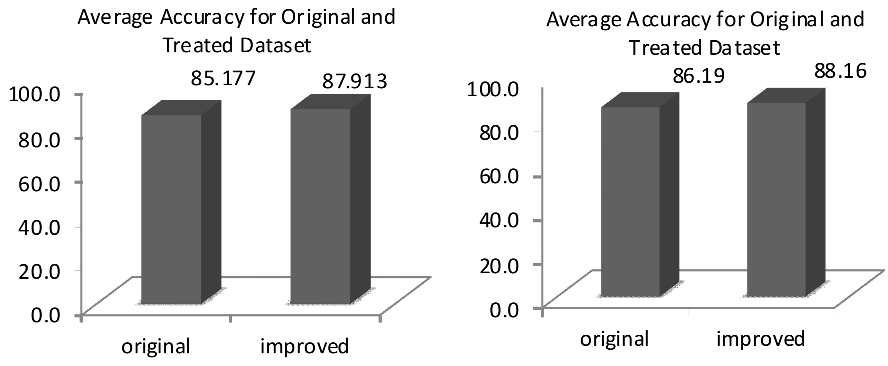


Fig. 3. Effects of data treatment on classification accuracies by classifiers count

As depicted in Fig. 3, for both case studies, the highest decreased in classification was identified for "w/o coordinates" treatment implying that when eliminating all body segment coordinates, 58 classifiers (in case study 1) and 55 classifiers (in case study 2) indicate weaker classification accuracy. Therefore, apparently the body segment coordinates are the important attributes for motion data classification.

The next effect is seen for case study 1; treatment without starting time and removal of missing value mark great improvements in the accuracy as compared to others treatments. Dataset without starting time contributes about 39% of classifiers showing improvements in classification accuracy. Approximately, more than 50% of the classifier counts showing an increase in accuracy. On the other hand, the accuracies for about 50% of classifiers remain constant when a single attribute either the speed or the distance was removed. This effect indicates that both the speed and distance attributes do not play a vital role to distinguish the study data into their belonging classes. Meanwhile in case study 2, the imputation effect on the identified missing values boosts the performances of the classification accuracies. About 64.5% of classifiers indicate an increase in the classification accuracy.

The overall findings show that combined results from randomization and attribute elimination treatments, indicate the good classification enhancement i.e. randomizing the class orders and removing the existing missing values as well as both the 'distance and speed' attributes eliminated for case study 1. The imputation treatment to predict missing values works well for case study 2 but not in case 1. This shows that imputing missing values is rather a problem-based approach. Fig. 4 compares the average classification accuracies for the original data and the treated dataset via the best treatment.



(a) (b)

Fig. 4. Average classification accuracy for the original untreated data and treated dataset on best treatment for (a) case study 1 and (b) case study 2

The key observation here is that there is an increase of approximately 3.28% and 1.97% in average classification accuracy for treated dataset of case study 1 and case study 2 respectively as compared to the original data. It is noteworthy from the data treatments that the body segment coordinates are the essential attributes in bipedal motion classifications. Meanwhile, the time spent on collecting distance and speed attributes is simply redundant as the classification performance is not ideal, though, having these additional information in case 1.

4 Conclusion

This study demonstrated the effects of different data treatments: randomization, attribute elimination and missing value imputation on classifying the bipedal motion data which has never been reported in the literatures.

The integration of randomization without distance and speed attributes treatment indicate the best classification enhancement showing 3.21% improvement in percentage accuracy compared to the original data. Meanwhile 2.29 % enhancement was observed on data imputation coupled with inclusion of data coordinates and randomization of case 2. The main advantage from having the data treatments is that it simplifies classification performances on motion data without inclusive of redundant attributes like the distance and speed. The new findings showed that motion analysis classification is mainly governed by the coordinates of the body segment as the essential attributes to promote good classification accuracies. The data treatments on motion data put emphasis on extracting significant attributes to simplify data transformation works and yet promotes good classification performances.

Acknowledgments. The authors gratefully acknowledge the financial supports provided by Universiti Sains Malaysia (USM) under the Research University (RUI) Scheme (1001/PMEKANIK/814188) and the Short Term Grant Scheme (304/PMEKANIK/60311053).

References

1. Slocum, D.B., James, S.L.: Biomechanics of Running. JAMA 205(11), 721–728 (1968)
2. McGeer, T.: Passive Bipedal Running. Proceedings of the Royal Society of London, B. Biological Sciences 240(1297), 107–134 (1990)
3. Rubenson, J., Lloyd, D.G., Besier, T.F., Heliams, D.B., Fournier, P.A.: Running in Ostriches (Struthio Camelus): Tthree-dimensional Joint Axes Alignment and Joint Kinematics. Journal of Experimental Biology 210(14), 2548–2562 (2007)
4. Schaller, N.U., D'Août, K., Villa, R., Herkner, B., Aerts, P.: Toe Function and Dynamic Pressure Distribution in Ostrich Locomotion. The Journal of Experimental Biology 214(7), 1123–1130 (2011)
5. Hwang, S.J., Choi, H.S., Kim, Y.H.: Motion Analysis Based on A Multi-segment Foot Model in Normal Walking. In: 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, IEMBS 2004 (2004)
6. Abdolahi, B., Ghasemi, S., Gheissari, N.: Human Motion Analysis Using Dynamic Textures. In: 2012 16th CSI International Symposium on Artificial Intelligence and Signal Processing (AISP) (2012)
7. Zhou, F., De la Torre, F., Hodgins, J.: Hierarchical Aligned Cluster Analysis for Temporal Clustering of Human Motion. Pattern Analysis and Machine Intelligence 99 (2012)
8. Maki, A., Perbet, F., Stenger, B., Cipolla, R.: Detecting Bipedal Motion from Correlated Probabilistic Trajectories. Pattern Recognition Letters (2013), http://dx.doi.org/10.1016/j.patrec.2012.12.019
9. Kadu, H., Kuo, M., Kuo, C.C.J.: Human Motion Classification and Management Based on Mocap Data Analysis. In: Proceedings of the 2011 Joint ACM Workshop on Human Gesture and Behavior Understanding, Scottsdale, Arizona, USA (2011)
10. Endres, F., Hess, J., Burgard, W.: Graph-Based Action Models for Human Motion Classification. In: Proceedings of ROBOTIK 2012, 7th German Conference (2012)
11. Han, S., Lee, S., Peña-Mora, F.: Comparative Study of Motion Features for Similarity-Based Modeling and Classification of Unsafe Actions in Construction. J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000339 (2013)
12. Kwak, N.J., Song, T.: Human Action Classification and Unusual Action Recognition Algorithm for Intelligent Surveillance System. In: Kim, K.J., Chung, K.-Y. (eds.) IT Convergence and Security 2012. LNEE, vol. 215, pp. 797–804. Springer, Heidelberg (2013)
13. Capodiferro, L., Costantini, L., Mangiatordi, F., Palloti, E.: Data Preprocessing to Improve SVM Video Classification. In: 2012 10th International Workshop on Content-Based Multimedia Indexing (CBMI), Annecy, pp. 1–4 (2012)
14. Saravanan, D., Srininvasan, S.: Video Image Retrieval Using Data Mining Techniques. Journal of Computer Applications (JCA) V(1), 39–42 (2012) ISSN: 0794-1925
15. Narasimhan, H., Tripuraribhatla, R., Easwarakumar, K.S.: Fuzzy Inference Based Data Preprocessing for VBR Video Traffic Prediction. Research Gate (2010), doi:10.1109/IMSAA.2010.5729394
16. Chan, C.K., Loh, W.P., AbdRahim, I.: Data Elimination cum Interpolation for Imputation: A Robust Preprocessing Concept for Human Motion Data. Procedia-Social and Behavioral Sciences 91, 140–149 (2013)
17. Youtube. Sports Science: Dennis Northcutt vs. Ostrich [Video file], https://www.youtube.com/watch?v=2WcMRpozO4s (February 1, 2012) (retrieved)

Experimental Analysis of Firefly Algorithms for Divisive Clustering of Web Documents

Athraa Jasim Mohammed[1,2], Yuhanis Yusof[1], and Husniza Husni[1]

[1] School of Computing, College of Arts and Sciences, Universiti Utara Malaysia,
06010 Sintok, Kedah, Malaysia
s94734@student.uum.edu.my, {yuhanis,husniza}@uum.edu.my
[2] Information and Communication Technology Center,
University of Technology, Baghdad, Iraq
autoathraa@yahoo.com

Abstract. This paper studies two clustering algorithms that are based on the Firefly Algorithm (FA) which is a recent swarm intelligence approach. We perform experiments utilizing the Newton's Universal Gravitation Inspired Firefly Algorithm (GFA) and Weight-Based Firefly Algorithm (WFA) on the 20_newsgroups dataset. The analysis is undertaken on two parameters. The first is the alpha (α) value in the Firefly algorithms and latter is the threshold value required during clustering process. Results showed that a better performance is demonstrated by Weight-Based Firefly Algorithm compared to Newton's Universal Gravitation Inspired Firefly Algorithm.

Keywords: Firefly algorithm, text clustering, divisive clustering.

1 Introduction

Clustering partitions a dataset of an unlabeled objects into particular number of clusters [1]. Clustering techniques is a powerful mechanism that engaged in many disciplines such as medicine [2], management [3] and finance [4]. These techniques can be divided into two categories based on mechanism of producing clusters. These categories are the partitional clustering and hierarchical clustering. Partitional clustering technique produces a single level of clusters; hence, it is also known as flat clustering [5]. An example of a popular partitional technique is the K-means, which is known due to its simplicity of implementation and efficiency [6]. However, the search process in this method is deterministic, and this may lead to local optima issues [7].

On the other hand, the hierarchical clustering technique generates multi-level of clusters. The agglomerative and divisive are two types of hierarchical clustering technique. The agglomerative hierarchical clustering technique merges a set of clusters based on some criterion such as similarity or distance between clusters. The divisive hierarchical clustering technique grouped a set of objects into specific clusters based on some partitional clustering techniques [8]. Bisect K-means method is the most known of divisive hierarchical clustering technique. This method constructs a hierarchy of clusters and at each level clusters are identified using the k-means method [9].

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_46, © Springer International Publishing Switzerland 2014

The problem in clustering is how can we group sets of objects into specific number of clusters? And how can we achieve the highest similarity between objects in a cluster while having the largest distance between the clusters? This problem can be represented as an optimization problem [7]. In optimization, we select the best solution from a group of available solutions [10]. Recently, various swarm intelligent optimization algorithm is applied to solve the problem of clustering such as the Ant Colony Optimization [11], Particle Swarm Optimization [8], Cuckoo Search Optimization Algorithm [12], Artificial Bee Colony [13] and Firefly Algorithm (FA) [14].

Firefly Algorithm is a nature inspired approach that is based on social insects behavior which are developed, by Yang in 2008 [15]. The flashing light of Fireflies and the attractiveness between them is the most important two factors in Firefly algorithm. The flashing light is related with the fitness of firefly and the attractiveness is related with the distance between two fireflies. The Firefly Algorithm has two important facts; automatic subdivision into subgroups and the capability of multi-modality [16, 17].FA has been successfully applied to solve hard optimization problems such as Speech Recognition [18], Image processing [19], Anomaly detection [20], Economic Dispatch problems [21], Mobil Network [22], Discrete Optimization problems [23] and Data Clustering [7, 14].

In [7], the researchers studied Firefly Algorithm in an unsupervised clustering. The experimental results demonstrate that the proposed Firefly algorithm outperformed particle swarm optimization (PSO) and differential evolution (DE) algorithms in achieving optimal solutions with a better convergence rate. On the other hand, the work reported in [14] explores Firefly Algorithm in supervised environment. The experimental results prove that the suggested Firefly Algorithm performs better than two swarm algorithms which are the particle swarm optimization (PSO) and Artificial Bee Colony (ABC).

In this paper, performances of two clustering algorithms that are based on Firefly Algorithm are compared. They are the Weight-Based Firefly Algorithm [24] and Newton's Universal Gravitation Inspired Firefly Algorithm [25]. We study the performance of these variants of Firefly clustering algorithms based on two factors which are the alpha (α) parameter and the threshold value utilized for constructing clusters. In the undertaken experiments, different values are utilized as the threshold and alpha (α). The results of these experiments are measured using three performance metrics which are Purity, F-measure and Entropy.

The remainder of the paper is structured as follows: In section 2, we provide the Weight-based Firefly Algorithm (WFA) approach [24] while section 3 includes the Newton's Universal Gravitation Inspired Firefly Algorithm approach [25]. Section 4 includes the parameter setting of algorithms. Respectively, section 5 and 6 present the utilized dataset and the performance measurements. The analysis of comparison is presented in section 7 and this is followed by the conclusion in section 8.

2 Weight-Based Firefly Algorithm

The Weight-Based Firefly Algorithm (WFA) [24] identifies the initial cluster center using an objective function which is based on the total weight of a document. It starts with a single cluster that contains all documents and works as in the following steps:

1. The initial number of fireflies is equal to the number of documents.
2. Find the total weight of each document by calculating the summation of all terms in the document. The utilized function is as in equation 1.

$$total\ weight_{d_j} = \sum_{i=1}^{m} tf - idf_{t_i,d_j} \tag{1}$$

Where, m is the number of terms.

3. Assign the total weight of a document as the light intensity of the Firefly.
4. Compare the light intensity between two fireflies; firefly with the brightest light attracts the less ones. Hence, moving the less bright firefly to the brightest one (as shown in equation 2). This will increase the light intensity of the brightest firefly (as shown in equation 3) based on attractiveness between Fireflies (as shown in equation 4 [15]).

$$X^{\ i} = X^i + \beta * (X^j - X^i) + \alpha \tag{2}$$

$$I^j = I^j + \beta \tag{3}$$

$$\beta = \beta_0 exp^{(-\Upsilon r_{ij}{}^2)} \tag{4}$$

Where, Xi, Xj are the positions of two fireflies, β is the attractiveness, α is a value between (0, 1).

5. Repeat step 4 until the number of iteration is reached.
6. Rank the documents to identify the center for a cluster. Document (i.e firefly) with the brightest light is selected as the centroid. Then, construct two clusters; the first cluster contains documents that are similar to the newly identified centroid while the second includes the dissimilar ones. Similarity between centroid and a document is based on cosine similarity and is supposed to be greater than a pre-defined threshold.
7. Repeat step 6 to find new centroids for the second cluster and the process is replicated until all documents are clustered.

3 A Newton's Universal Gravitation Inspired Firefly Algorithm

The Gravitation Inspired Firefly Algorithm [25] tries to find the center of cluster by using an objective function that is based on the force between each documents and center of cluster. The force between two documents relies on three variables; the similarity, distance and the weight of each document (i.e the total weight of document). It starts with a single cluster that contains all documents and works as in the following steps:

1. The initial number of fireflies is equal to the number of documents.
2. Construct a similarity matrix using equation 5.

$$similarity = \sum_{j=1}^{m} (D_j * V_j) \tag{5}$$

3. Construct a distance matrix using equation 6 [28].

$$Euclidean\ distance(D_j, D_j) = \sqrt[2]{(D_i - D_j)^2} \tag{6}$$

4. Calculate the total weight of each document using equation 1.
5. Construct a matrix containing the force between two documents based on the matrixes constructed in steps 2, 3 and 4. The force between two documents is as in equation 7 [25].

$$F(D_i, D_j) = similarity(D_i, D_j) * \frac{D_i * D_j}{distance(D_i, D_j)^2} \tag{7}$$

6. Find the total force of each document and assign the total force of each document as the light intensity of a Firefly.
7. Compare the brightness between two fireflies. The brightest one attracts the less bright firefly, hence moving the latter to the brightest firefly (use equation 2 and 4).
8. Update the distance, force and total force matrixes.
9. Repeat step 7 and 8 until the number of iteration is reached.
10. Rank to find documents that have the brightest light and represent it as the centroid.
11. Construct two clusters where the first cluster contains documents that are similar to the centroid while the second cluster includes dissimilar documents. Determination of similarity is based on a threshold value where only documents with similarity value greater than a pre-defined threshold will be assigned in the first cluster.
12. Repeat step 10 and onwards on the second cluster to find new centroids. The process is continued until the last document is clustered.

4 Parameter Setting of Algorithms

In executing the WFA and GFA, we require two pre-defined values; the alpha (α) value is required in moving the fireflies while the threshold value is utilized in identifying similarity between a document and the identified centroid. Different parameter settings produce different quality clusters. Based on literature [16], the alpha value is range between 0 and 1. Hence, in this paper, we utilize values that is below than 0.5 and above than 0.5. The undertaken experiments utilizes alpha = 0.2 and alpha = 0.7. In addition, we test three values of threshold (0.15, 0.175, and 0.2) in the experiments.

5 Dataset

The two comparative algorithms are tested on a benchmark dataset which is the 20_newsgroups dataset [26]. We select 300 documents from 3 different classes where each class contains 100 documents. Table 1 contains description of the documents from the 20_newsgroups dataset.

Table 1. Description of 20_newsgroups Dataset

Dataset Topics	No. of Documents	Classes	Total No. of Documents	Total No. of Classes	No. of Terms
Comp.sys.mac.hardware	100	1	300	3	2214
Rec.sport.baseball	100	1			
Sci.electronic	100	1			

6 Performance Measurements

We use three performance metrics to measure the quality of clusters; Purity which measures the cluster that include only one class (as shown in equation 8), F-measure indicates the accuracy of the algorithm (as shown in equation 9,10) and Entropy provides the distribution of classes in each cluster (as shown in equation 11,12) [24, 25, 27].

$$Purity = \sum_{\Omega_k \in \{\Omega_1,\ldots,\Omega_c\}} \frac{\mathrm{Max_k}\ |\Omega_k \cap C_j|}{N} \tag{8}$$

$$Total\ F - measure = -\sum_{k=1}^{C} \frac{|\Omega_k|}{N} * max\big(F(\Omega_k)\big) \tag{9}$$

$$F(\Omega_k) = \max_{C_j \in \{C_1,\ldots,C_k\}} \left(\frac{2 * R(\Omega_k, C_j) * P(\Omega_k, C_j)}{R(\Omega_k, C_j) + P(\Omega_k, C_j)} \right) \tag{10}$$

$$H(j) = -\sum_{k=1}^{C} \frac{|\Omega_k \cap C_j|}{|C_j|} log \frac{|\Omega_k \cap C_j|}{|C_j|} \tag{11}$$

$$H = -\sum_{j=1}^{k} \frac{H_j * |C_j|}{N} \tag{12}$$

7 Analysis of the Comparison

In Figure 1 and 2, we show the relation between the utilized threshold (which is used to construct clusters in algorithm) and the performance measurement of Purity in two comparative algorithms; Weight-Based Firefly Algorithm (WFA) and Newton's Universal Gravitation Inspired Firefly Algorithm (GFA). The first analysis (refer Figure 1) is based on one iteration with α=0.7 and the second (refer Fig. 2) is based on α=0.2. Illustration in Fig. 1 shows that the purity of WFA is better than GFA for the utilized threshold values with α = 0.7. It is also learned that both algorithms performed the best (for purity) at threshold = 0.2

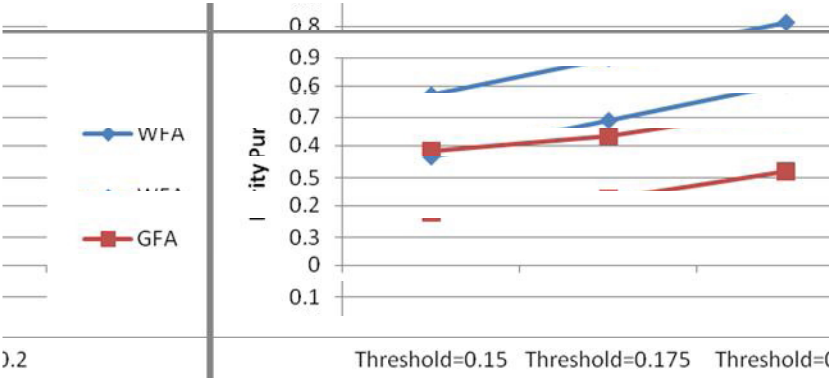


Fig. 1. Clustering Purity at iteration=1 and α =0.7

Data in Fig. 2 indicates that the purity of WFA is better than GFA for threshold values of 0.15 and 0.2. However they perform the best at 0.175.

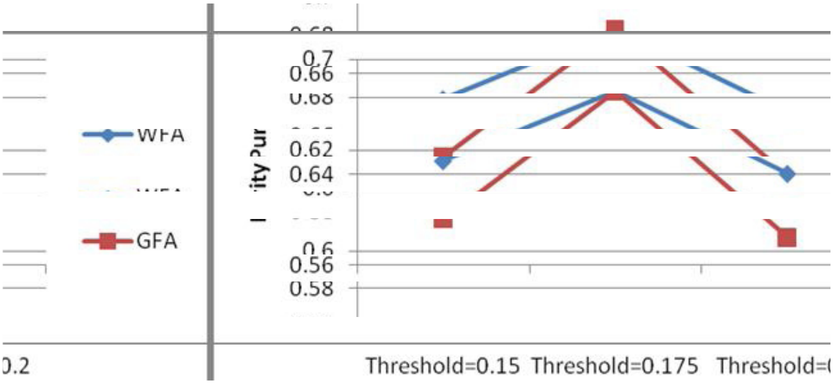


Fig. 2. Clustering Purity at iteration=1 and α =0.2

In Fig. 3 and 4, we can see the relation between the threshold and the quality of clustering which is F-measure in two comparative algorithms; WFA and GFA in one generation iteration and alpha is (0.7 and 0.2). In Figure 3, when α= 0.7, the value of the F-measure for WFA is better than GFA for all utilized threshold values. For both algorithms, it is noted that the F-measure are higher when the threshold is small (for example 0.15).

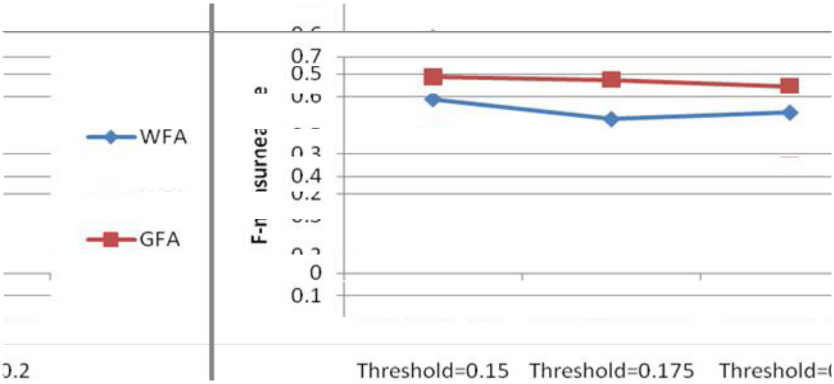


Fig. 3. F-measure at iteration=1 and α =0.7

Contradict to the one in Figure 3, when α=0.2, the F-measure obtained by GFA is better than WFA for the three threshold values. Such a result is illustrated in Figure 4. The utilized threshold= 0.175 produced a better F-measure in GFA but vice versa for WFA. Nevertheless, both algorithms produced less value when threshold=0.2.

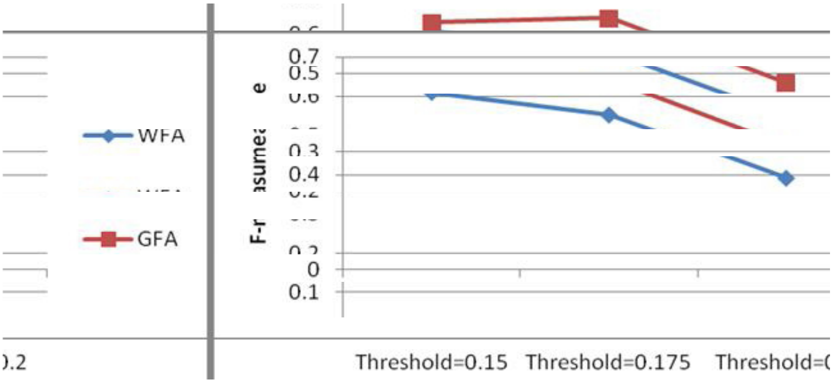


Fig. 4. F-measure at iteration=1 and α =0.2

In Fig. 5 and 6, we show the relation between the distribution of documents in clusters (Entropy) and the threshold in two comparative algorithms; WFA and GFA in one generation iteration and alpha is (0.7 and 0.2). In Fig. 5 when the α= 0.7, The Entropy value in WFA is lower than GFA in all thresholds which it means better result. From the figure, we learned that a larger threshold value (for example 0.2) produces better result for both algorithms.

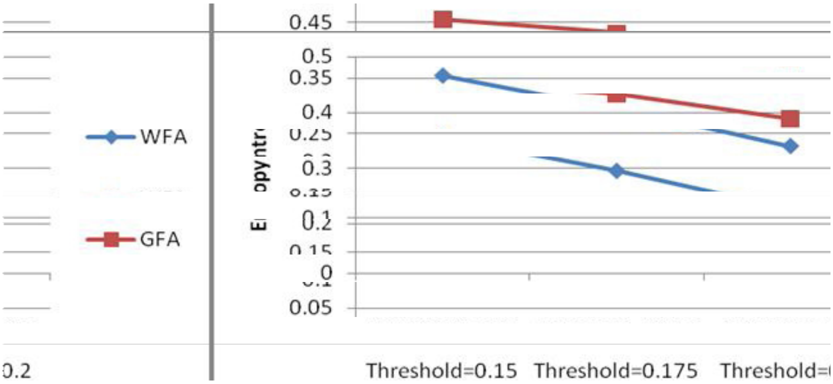


Fig. 5. Clustering Entropy at iteration=1 and α =0.7

In Fig. 6 when α= 0.2, the Entropy value in WFA is lower than GFA in all utilized thresholds, hence meaning that the WFA is a better algorithm. From the figure, we learned that the best result was obtained while using 0.2 for WFA and 0.175 for GFA.

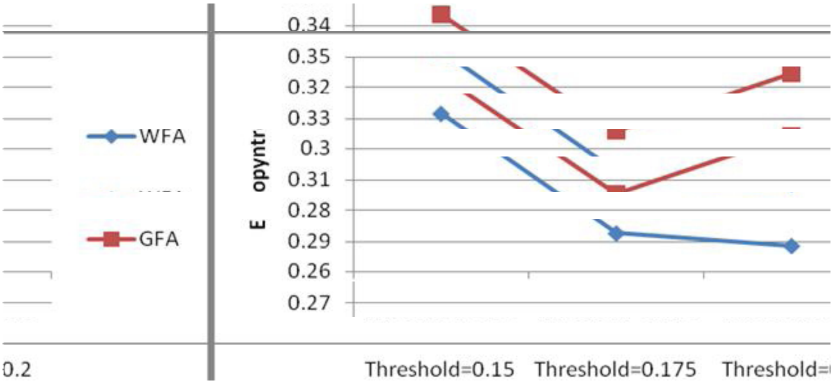


Fig. 6. Clustering Entropy at iteration=1 and α =0.2

The experimental results indicate that the threshold 0.2 is suitable for two algorithms to increase purity and decrease Entropy when the value of alpha is 0.7. On the other hand, the suitable threshold value is 0.175 when alpha (α) is 0.2. Furthermore, the utilized threshold of value 0.15 increases the F-measure for both values of alpha (0.7 and 0.2). It is learned that the WFA produces a better result when alpha is 0.7 but GFA is best at alpha = 0.2. A good clustering is when the algorithm produces a high purity and F-measure values while having low Entropy [27]. Hence, from the experimental results, we concluded that WFA outperformed GFA in 20Newsgroup dataset in Purity and Entropy metrics in all value of threshold and alpha (α) parameter.

8 Conclusions

In this paper, we presented the experimental results of two divisive clustering algorithms that are based on Firefly Algorithm (FA). In particular, we compare the Weight-Based Firefly Algorithm with Newton's Universal Gravitation Inspired Firefly Algorithm. The effect of factors, such as the change in α parameter and the threshold value required for constructing clusters, on the quality of clustering is compared between the two algorithms. Experimental results on 20 news group dataset indicate that WFA produced higher value in two metrics; purity and F-measure, and lower value of Entropy metrics in the different threshold and alpha (α) parameter.

References

1. Tan, P.N., Steinbach, M., Kumar, V.: Introduction to Data Mining Clustering, 1st edn. Pearson (2014)
2. Chaira, T.: A Novel Intuitionistic Fuzzy C means clustering Algorithm and its application to Medical Image. Applied Soft Computing 11(2), 1711–1717 (2011)
3. Ngai, E.W.T., Xiu, L., Chau, D.C.K.: Application of Data Mining Technique in Customer Relationship management: A literature review and Classification. Expert Systems with Applications 36(2), 2592–2602 (2009)
4. Zhang, D., Zhou, L.: Discovering Golden nuggets: Data Mining in Financial Application. IEEE Transactions on Systems,Man, and Cybernetics, Part C: Application and Reviews 34(4), 513–522 (2004)
5. Luo, C., Li, Y., Chung, S.M.: Text Document Clustering based on Neighbors. Data and Knowledge Engineering 68(11), 1271–1288 (2009)
6. Jain, A.K.: Data Clustering: 50 years beyond K-means. Pattern Recognition Letters 31(8), 651–666 (2010)
7. Banati, H., Bajaj, M.: Performance Analysis of Firefly Algorithm for Data Clustering. International Journal Swarm Intelligence 1(1) (2013)
8. Feng, L., Qiu, M.H., Wang, Y.X., Xiang, Q.L., Yang, Y.F., Liu, K.A.: A Fast Divisive Clustering Algorithm Using an Improved Discrete Particle Swarm Optimizer. Pattern Recognition Letters 31(11), 1216–1225 (2010)
9. Kashef, R., Kamel, M.S.: Enhanced Bisecting K-means Clustering using Intermediate Cooperation. Pattern Recognition 42(11), 2557–2569 (2009)
10. Rothlauf, F.: Design of Modern Heuristics Principles and Application. Springer, Heidelberg (2011)
11. He, Y., Hui, S.C., Sim, Y.: A Novel Ant-based Clustering Approach for Document Clustering. In: Ng, H.T., Leong, M.-K., Kan, M.-Y., Ji, D. (eds.) AIRS 2006. LNCS, vol. 4182, pp. 537–544. Springer, Heidelberg (2006)
12. Zaw, M.M., Mon, E.E.: Web Document Clustering using Cuckoo Search Clustering Algorithm based on Levy Flight. International Journal of Innovation and Applied Studies 4(1), 182–188 (2013)
13. Karaboga, D., Ozturk, C.: A Novel Clustering Approach: Artificial Bee Colony (ABC) Algorithm. Applied Soft Computing 11(1), 625–657 (2011)
14. Senthilnath, J., Omkar, S.N., Mani, V.: Clustering Using Firefly Algorithm: Performance Study. Swarm and Evolutionary Computation 1(3), 164–171 (2011)

15. Yang, X.S.: Nature-inspired Metaheuristic Algorithms, 2nd edn. Luniver Press, United Kingdom (2011)
16. Yang, X.S., He, X.: Firefly Algorithm: Recent Advances and Applications. Int. J. Swarm Intelligence 1(1), 36–50 (2013)
17. Fister, I., Fister Jr., I., Yang, X.S., Brest, J.: A Comprehensive Review of Firefly Algorithms 13, 34–46 (2013)
18. Hassanzadeh, T., Faez, K., Seyfi, G.: A Speech Recognition System Based on Structure Equivalent Fuzzy Neural Network Trained by Firefly Algorithm. In: International Conference on Biomedical Engineering (ICoBE), pp. 63–67. IEEE (2012)
19. Horng, M.H., Jiang, T.W.: Multilevel Image Thresholding Selection based on the Firefly Algorithm. In: 7th International Conference on Ubiquitous Intelligence & Computing and 7th International Conference on Autonomic & Trusted Computing (UIC/ATC), pp. 58–63. IEEE (2010)
20. Adaniya, M.H.A.C., Abrão, T., Proença Jr., M.L.: Anomaly Detection Using Metaheuristic Firefly Harmonic Clustering. Journal of Networks 8(1), 82–91 (2013)
21. Yang, X.S., Hosseini, S.S.S., Gandomi, A.H.: Firefly Algorithm for solving non-convex economic dispatch problems with valve loading effect. Applied Soft Computing 12(3), 1180–1186 (2012)
22. Bojic, I., Podobnik, V., Ljubi, I., Jezic, G., Kusek, M.: A self-optimizing mobile network: Auto-tuning the network with firefly-synchronized agents. Information Sciences 182(1), 77–92 (2012)
23. Sayadi, M.K., Hafezalkotob, A., Naini, S.G.J.: Firefly-inspired algorithm for discrete optimization problems: An application to manufacturing cell formation. Journal of Manufacturing Systems 32(1), 78–84 (2013)
24. Mohammed, A.J., Yusof, Y., Husni, H.: Weight-Based Firefly Algorithm for Document Clustering. In: Herawan, T., Deris, M.M., Abawajy, J. (eds.) Proceedings of the First International Conference on Advanced Data and Information Engineering (DaEng 2013). LNEE, vol. 285, pp. 259–266. Springer, Heidelberg (2014)
25. Mohammed, A.J., Yusof, Y., Husni, H.: A Newton's Universal Gravitation Inspired Firefly Algorithm for Document Clustering. In: Jeong, H.Y., Yen, N.Y., Park, J.J(J.H.) (eds.) Advanced in Computer Science and its Applications. LNEE, vol. 279, pp. 1259–1264. Springer, Heidelberg (2014)
26. 20 Newsgroup Data Set (2006), `http://people.csail.mit.edu/20Newsgroup/`
27. Murugesan, K., Zhang, J.: Hybrid Bisect K-means Clustering Algorithm. In: IEEE International Conference on Business Computing and Global Informatization (BCGIN), pp. 216–219. IEEE (2011)
28. Hassanzadeh, T., Meybodi, M.R.: A New Hybrid Approach for Data Clustering Using Firefly Algorithm and K-means. In: Proceedings of the 16th IEEE CSI International Symposium on Artificial Intelligence and Signal Processing (AISP), pp. 007 – 011 (2012)

Extended Naïve Bayes for Group Based Classification

Noor Azah Samsudin[1] and Andrew P. Bradley[2]

[1] Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia,
86400 Parit Raja, Batu Pahat, Johor, Malaysia
{azah,mmustafa,shamsulk}@uthm.edu.my
[2] School of Information Technology and Electrical Engineering
The University of Queensland, 4067 QLD, Australia
bradley@itee.uq.edu.au

Abstract. This paper focuses on extending Naive Bayes classifier to address group based classification problem. The group based classification problem requires labeling a group of multiple instances given the prior knowledge that all the instances of the group belong to same unknown class. We present three techniques to extend the Naïve Bayes classifier to label a group of homogenous instances. We then evaluate the extended Naïve Bayes classifier on both synthetic and real data sets and demonstrate that the extended classifiers may be a promising approach in applications where the test data can be arranged into homogenous subsets.

Keywords: group based classification, Naïve Bayes, classification.

1 Introduction

Group based classification (GBC) problem is about labeling a group of multiple instances with the prior knowledge that all the instances in the group belong to same but unknown class [1, 2]. The GBC problem arises in various applications in which there is a need to determine class membership of an object represented by multiple instances such as in cervical cancer screening [3-6] and plant species classification problems discussed in [2].

This paper introduces three techniques to extend Naive Bayes [7] classifier to label a group of instances. We then evaluate the extended Naive Bayes classifiers on both synthetic and real data sets and demonstrate that the extended Bayes classifiers may be a promising approach in applications where the test data can be arranged into homogenous subsets. The performances of the proposed classifiers are compared with the conventional Naive Bayes classifier and another GBC technique, namely F-test based classifier [2].

The rest of the paper is organized as follows. Section 2 formally reviews the property of Naïve Bayes classifier. Section 3 describes the three techniques that we implemented for solving the GBC problem. Section 4 presents the data sets and our experiment methodology. Section 5 presents the results of various techniques and finally provides some concluding remarks.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_47, © Springer International Publishing Switzerland 2014

2 Naïve Bayes Classifier

The principal idea of Naïve Bayes classifier originates from Bayes' Theorem [7]. In a classification problem, we are given a data set consisting of N instances and their associated class labels, such as $D = \{(\mathbf{x}^1{}_, c^1), (\mathbf{x}^2{}_, c^2),\ldots, (\mathbf{x}^N{}_, c^N)\}$. Each instance $\mathbf{x}$ is represented by n-dimensional measurements, which are also known as feature vectors—that is, $\mathbf{x} = (f_1, f_2, \ldots, f_n)$. The n-dimensional measurements presented in each instance are obtained from a set of features, $F_1, F_2, \ldots, F_n$. c^N is a class label for $\mathbf{x}^N$, where l belongs to a set of class labels, such that $l=1, \ldots,L, L > 1$.

The aim of a classification problem is to determine the class membership of a single instance, $\mathbf{x}$. Applying Bayes' Theorem, the class membership is determined according to the class that has the highest posterior probability, conditioned on $\mathbf{x}$. Using the Bayes' Theorem, the principal idea of the classification problem can be written as:

$$P(c_l \mid \mathbf{x}) = \frac{P(\mathbf{x} \mid c_l)P(c_l)}{P(\mathbf{x})} \tag{1}$$

As $P(\mathbf{x})$ is constant for all classes, $P(\mathbf{x}|c_l)P(c_l)$ therefore needs to be maximised. If the prior probabilities $P(c_l)$ are unknown, the common assumptions are that the classes are equally likely, $P(c_1) = P(c_2) = \ldots = P(c_L)$, and we would therefore maximise $P(\mathbf{x}|c_l)$. That is, the classifier labels $\mathbf{x}$ belongs to class c_i only if:

$$P(c_i \mid \mathbf{x}) > P(c_j \mid \mathbf{x}) \text{ for } 1 \le j \le L, j \ne i$$

Clearly, a class label is determined according to the most likely possible classifications. Note that class prior probabilities can be estimated using the number of instances of class c^l presented in the given data set. A class label c for a single instance $\mathbf{x}$ using Equation (1) is given as:

$$c = \arg\max_{l=1..L} P(c_l)P(\mathbf{x} \mid c_l) \tag{2}$$

where $P(c^l)$ is the prior probability. Due to the naïve independence assumption in Naïve Bayes, all features $F_1, F_2, \ldots, F_n$ are conditionally independent given a class label. Therefore, $P(\mathbf{x}|c^l)$ can be decomposed into a product of n terms, one term for each feature, such as:

$$P(\mathbf{x} \mid c_l) = \prod_{i=1}^{n} P(\mathbf{x} = f_i \mid c_l) \tag{3}$$

Thus, the Naïve Bayes classification rule to determine class label c for an instance $\mathbf{x}$ can be defined as:

$$c = \arg\max_{l=1..L} P(c_l)\prod_{i=1}^{n} P(\mathbf{x} = f_i \mid c_l) \tag{4}$$

Given a data set *D* with *N* instances, if the prior probability $P(c_l)$ is unknown, we can estimate $P(c_1)$ for each class *l*, $\hat{P}(c_l) = \frac{N_l}{N}$, N_l is the number of instances of class *l*. Assuming that the measurements of all *n* features follow the Gaussian distribution, we can estimate the conditional probability for $P(\mathbf{x}= f_i \mid c^l)$ as:

$$P(\mathbf{x} = f_i \mid c_l) = g(f_i\ ; \mu_i, \sigma_i), \text{ e.g.}$$

$$g(f_i\ ; \mu_i, \sigma_i) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{ - \frac{(f - \mu)^2}{2\sigma^2} \right\} \tag{5}$$

The mean μ_i and the standard deviation σ_i are estimated using the measurements of features in the data set *D*.

3 Extended Naïve Bayes Classifier

We propose to extend the Naïve Bayes classifier to address the GBC problem as follows:

$$P(c_l \mid \mathbf{X}_{\mathrm{TE}}) = \frac{P(\mathbf{X}_{\mathrm{TE}} \mid c_l) P(c_l)}{P(\mathbf{X}_{\mathrm{TE}})} \tag{6}$$

given that $\mathbf{X}_{\mathrm{TE}}$ is a group of N_{TE} instances—for example, $\mathbf{X}_{\mathrm{TE}} = \{\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^{N_{\mathrm{TE}}}\}$, in which these N_{TE} instances belong to the same unknown class. The GBC approach clearly emphasises that the class label decision is dependent not just on a single instance, but on a group of instances. Note that our GBC aims to determine class label ***c*** for a group of instances. This is different from determining *c* for a single instance **x** presented in Equation (5). Assuming the features of the instances are independent, we propose three approaches to extend the Naïve Bayes classifier to implement the GBC:

1. Naïve Bayes (Voting) (NBV): We estimate posterior probability for every *k*-th instance, $\mathbf{x}^k$ in $\mathbf{X}_{\mathrm{TE}}$. Based on the posterior probability estimation, the class label for every *k*-th instance c_l^k is determined—for example, $c_l^k = \arg\max_{l=1\ldots L} p(c_l \mid \mathbf{x}^k)$. As the Bayes classifier outputs a posterior probability, this voting is actually a threshold at 0.5—that is, for a two-class problem, *i* and *j*, $p(c^i|\mathbf{x}) + p(c^j| \mathbf{x}) = 1$. The total vote of every class label v_l is then determined from the instances in $\mathbf{X}_{\mathrm{TE}}$—for example, $v_l = \sum_{k=1}^{N_{TE}} c_l^k$. Finally, $\mathbf{X}_{\mathrm{TE}}$ is labelled with the class of the majority vote—for example, $\boldsymbol{c}_l = \arg\max_{l=1\ldots L} v_l$.

2. Naïve Bayes (Naive Pooling) (NBP): Like in NBV, the posterior probability is estimated for every instance **x** in $\mathbf{X}_{\mathrm{TE}}$. Unlike NBV, these instances are not labelled individually. Instead, combined probabilities are estimated for every class l, $p(C_l)$, using instances in $\mathbf{X}_{\mathrm{TE}}$—for example, $p(C_l) = \dfrac{\prod_{k=1}^{N_{TE}} p(c_l \mid \mathbf{x}^k)}{\prod_{k=1}^{N_{TE}} p(c_l \mid \mathbf{x}^k) + \prod_{k=1}^{N_{TE}} (1 - p(c_l \mid \mathbf{x}^k))}$. Finally, $\mathbf{X}_{\mathrm{TE}}$ is labelled with a class with maximum combined probability—for example, $\boldsymbol{c}_l = \underset{l=1\ldots L}{\arg\max}\, p(C_l)$. The combined probability is related—for example, $p(C_i) = 1\text{-}p(C_j)$, for $i \neq j$.
3. Our naïve assumption in NBP is that each instance in $\mathbf{X}_{\mathrm{TE}}$ is independent from each other, in which the assumption is contrary to our proposed GBC assumption. It is unlikely that individual instances in the same group (e.g. sample petals from the same plant species or cells in a slide) are independent; in fact, the GBC assumes the opposite.
4. Naïve Bayes (Direct Pooling) (DNP): We aim to implement the proposed group-based classifier, $P(c_l \mid \mathbf{X}_{\mathrm{TE}}) = \dfrac{P(\mathbf{X}_{\mathrm{TE}} \mid c_l) P(c_l)}{P(\mathbf{X}_{\mathrm{TE}})}$ presented in Equation (6) directly. That is, we want to use all instances in $\mathbf{X}_{\mathrm{TE}}$ as a group to observe similarity with every class training set $\mathbf{X}_l$. Again in DNP, we make the naïve assumption that features are independent in order to estimate the probability density function (PDF) for every class training set $p(\mathbf{X}_l|c_l)$ for l = 1,..., L. We also estimate the PDF for $\mathbf{X}_{\mathrm{TE}}$, $p(\mathbf{X}_{\mathrm{TE}})$. Upon obtaining PDFs for the training sets $p(\mathbf{X}_l|c_l)$ and the test set $p(\mathbf{X}_{\mathrm{TE}})$, the next step is to use these PDFs to estimate the posterior probability for $p(c_l|\mathbf{X}_{\mathrm{TE}})$. Here, we apply the Kolmogorov–Smirnov test (also known as the K–S test) to measure the differences between every class $p(\mathbf{X}_l|c_l)$ and $p(\mathbf{X}_{\mathrm{TE}})$ using the empirical cumulative distribution function (CDF)—for example, $\mathrm{CDF}_l = cdf(p(\mathbf{X}_l|c_l))$ and $\mathrm{CDF}_{\mathrm{XTE}} = cdf(p(\mathbf{X}_{\mathrm{TE}}))$. The K–S test results in a probability of differences between CDF_l and $\mathrm{CDF}_{\mathrm{XTE}}$ e.g. p_l = K-S_test (CDF_l, $\mathrm{CDF}_{\mathrm{XTE}}$). Note that we are estimating the p_l in a naïve manner—that is, one feature at a time. To accommodate n features, we therefore determine the combined probability value for every class l—for example, $\mathbf{p}_l = \prod_{i=1}^{n} p_l^i$. Finally, $\mathbf{X}_{\mathrm{TE}}$ is labelled with a class with minimum combined probability—for example, $\boldsymbol{c}_l = \underset{l=1\ldots L}{\arg\min}\, \mathbf{p}_l$.

Different to NBV, where an individual instance is assigned a class label prior to the group classification stage, in DNP, the class label is assigned directly to the group. Different to NBP, where the instances in the same group are assumed to be independent in contrast to the GBC assumption, in DNP, the assumption of our proposed GBC is not violated.

4 Experiment Methodology

The purpose of our experiments was to initially investigate the efficacy of GBC on both synthetic and real data sets. As all of our experiments involved approximately equal class priors, the error rate was thought to be an appropriate measure of classification performance. This was estimated using 10-fold cross-validation [7, 8]. In the experiments, all instances in the data sets were used, and each cross-validation partition (fold) was randomly selected in order to preserve prior class probability. For every class l, one partition was used as the test data, $\mathbf{X}_{\mathrm{TE}}$, and the remaining partitions as the training data, $\mathbf{X}_l$. However, to plot the classification error rate as a function of group size, all possible subsets of $\mathbf{X}_{\mathrm{TE}}$ larger than size three were evaluated. For every combination size, the error rate was estimated by dividing the number of misclassified instances by the total instances, N.

In all experiments, the performances of the GBC techniques were compared against the Naïve Bayes classifier, which was chosen because the synthetic data sets were normally distributed; thus, it should perform well. Note that for the Naïve Bayes classifier, only one instance from the group is presented to the classifier at a time. Conversely, for the GBC techniques, a group of instances is presented to the respective classifier so that once the group is labelled as belonging to a particular class, all instances in the group are classified as belonging to that class.

4.1 Synthetic Data

We conducted our experiments with commonly used Gaussian data sets—namely the I-I, I-Λ and I-4I data sets originally developed by Fukunaga [9]. Notably, each data set has a different level of 'difficulty', with calculated Bayes error rates of 10 per cent, 1.9 per cent and 9 per cent for I-I, I-Λ and I-4I respectively. Each data set consists of an eight-dimensional data vector with 1,000 samples per class. In these synthetic data sets, $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ are the mean vectors for class 1 and class 2 respectively. Meanwhile, $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$ are the corresponding covariance matrices for each class. The values of the $\boldsymbol{\mu}_l$ and $\boldsymbol{\lambda}_l$ are given in Table 1, where I_8 is the 8×8 identity matrix. For the I-Λ data set, $\boldsymbol{\mu}_2$and $\boldsymbol{\lambda}_2$ are provided in Table 2. As we are using 10-fold cross-validation each test partition, $\mathbf{X}_{\mathrm{TE}}$**,** consists of 100 instances per class. For every test partition all possible subsets *(groups)* of size three and above were evaluated. In this way, the proposed GBC techniques determined the class labels for variously sized subsets of the test data. We chose to have subsets of odd-numbered size to avoid tie voting in experiments with NBV approach.

Table 1. Synthetic data sets I-I, I-4I and I-Λ

Data set	μ_1	μ_2	λ_1	λ_2
I-I ($\mu_{1\neq}\ \mu_2$, $\lambda_{1=}\lambda_2$)	0	[2.56, 0,…,0]	I_8	
I-Λ ($\mu_{1\neq}\ \mu_2$, $\lambda_{1\neq}\lambda_2$)		[μ_1,…, μ_8] (Table 2)		[λ_1,…, λ_8] (Table 2)
I-4I ($\mu_{1=}\ \mu_2$, $\lambda_{1\neq}\lambda_2$)	0		I_8	$4I_8$

Table 2. Parameter values of the I-Λ data set

Dimension i	1	2	3	4	5	6	7	8
μ_i	3.86	3.10	0.84	0.84	1.64	1.08	0.26	0.01
λ_i	8.41	12.06	0.12	0.22	1.49	1.77	0.35	2.73

4.2 Iris Data

The iris data set is a collection of plant species from three classes: *Iris setosa*, *Iris versicolor* and *Iris virginica* [10, 11]. There are 50 samples from each class, and each sample is represented by measurements of four features: petal length, petal width, sepal length and sepal width. The data set is obtained from the UCI Machine Learning Repository (http://archive.ics.uci.edu). With 10-fold cross-validation, each test partition, $\mathbf{X}_{TE}$, consists of five (homogenous) instances of unknown class. Therefore, we classified every combination of the test set of size three to five instances.

5 Results

Figures 1–3 show the plots of error rates as a function of group size for each case of synthetic data. In all cases, the error rate for the GBC techniques approach zero as the group size increases. For all three synthetic cases, most of the proposed classifiers outperform the Naïve Bayes when the combination size is larger than seven. Indeed, it is interesting to note that an error rate of zero can be achieved with these data,

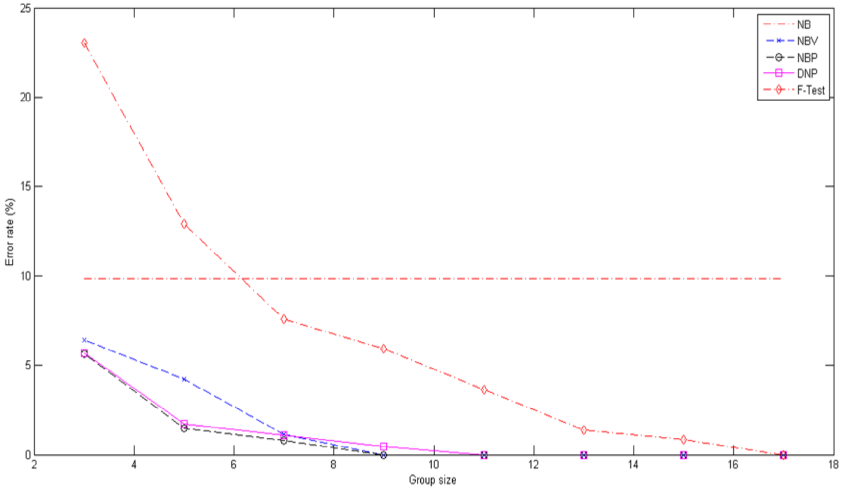


Fig. 1. Comparison of error rate (%): GBC techniques and Naïve Bayes for Case II

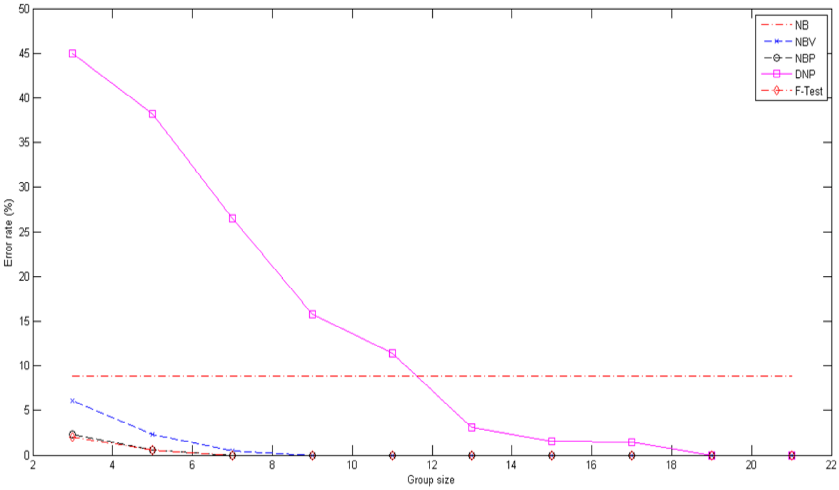


Fig. 2. Comparison of error rate (%): GBC techniques and Naïve Bayes for Case I-4I

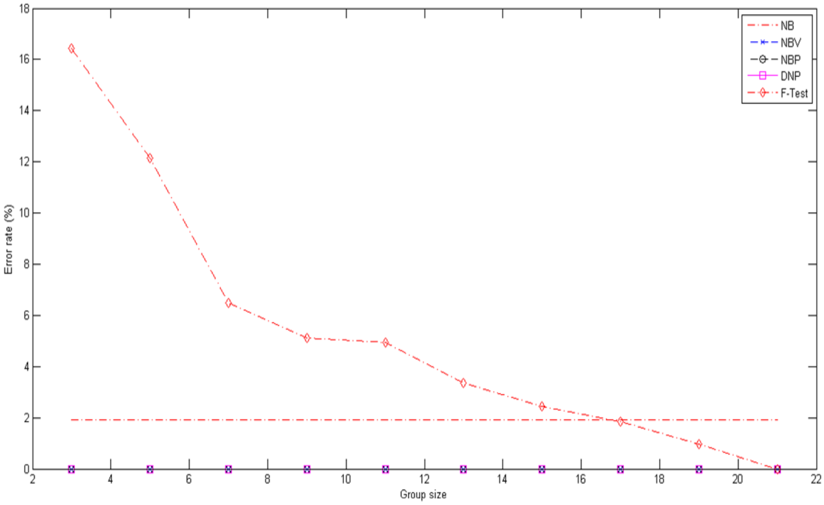


Fig. 3. Comparison of error rate (%): GBC techniques and Naïve Bayes for Case **I-Λ**

which by definition have overlapping class probability distributions and thus a non-zero Bayes error. This indicates the potential benefits of utilising the additional prior knowledge implicit to GBC—that is, that a group of test instances has the same but unknown class membership. The Iris data set is used as one potential practical application of GBC by arranging the test data into homogenous subgroups. The results in Figure 4 suggest that GBC outperforms the Naïve Bayes classifier when the group size is three or more. Clearly, GBC is benefitting from the prior knowledge that all test samples in the sub-group are homogeneous and should be given the same class label.

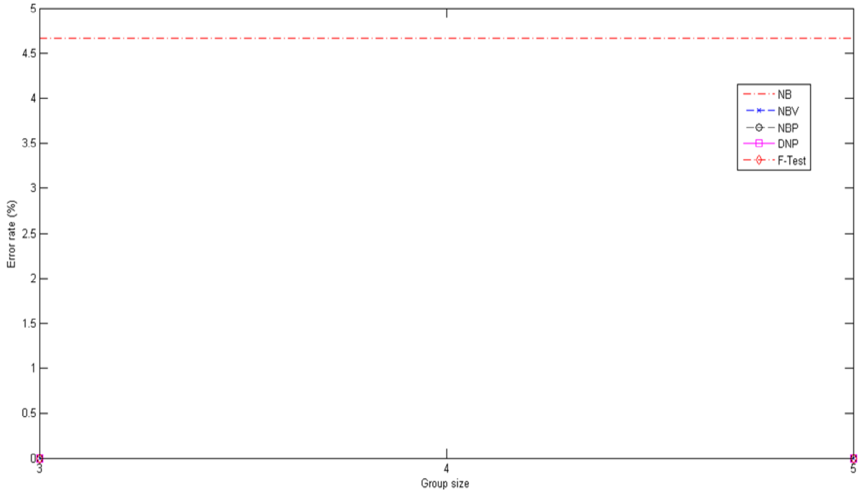


Fig. 4. Comparison of error rate (%): GBC techniques and Naïve Bayes for the Iris data set

6 Conclusions

In this paper, we presented the underlying concepts and rationale behind GBC. We developed and evaluated four GBC techniques, comparing them to individual based classification technique, the conventional Naïve Bayes classifier. In particular, three different ways of extending the Naive Bayes classifier to accumulate information about a group of test samples were presented: voting, naive pooling and direct pooling. The extended Naive Bayes classifiers were then evaluated for a variety of group sizes using both synthetic and real-world data, and their performances were evaluated in terms of average error rate. The results indicate that the proposed GBC techniques have the potential to outperform the Naive Bayes classification technique, especially as the (group) size of the test set increases. Clearly, these results indicate that the additional prior knowledge that a group of test samples belongs to the same but unknown class label can be effectively utilised to reduce misclassification problems.

References

1. Samsudin, N.A., Bradley, A.P.: Nearest neighbour group-based classification. Pattern Recognition 43, 3458–3467 (2010)
2. Samsudin, N.A., Bradley, A.P.: Group-based meta-classification. In: 19th International Conference on Pattern Recognition, IEEE, Tampa (2008)
3. Moshavegh, R., Bejnordi, B.E., Mehnert, A., Sujathan, K., Malm, P., Bengtsson, E.: Automated segmentation of free-lying cell nuclei in Pap smears for malignancy associated change analysis. In: 34th Annual International Conference of the IEEE EMBS, pp. 5372–5375 (2012)

4. Bengtsson, E.: Computerized cell image analysis: Past, present, and future. In: Bigun, J., Gustavsson, T. (eds.) SCIA 2003. LNCS, vol. 2749, pp. 395–407. Springer, Heidelberg (2003)
5. Nordin, B., Bengtsson, E.: Specimen analysis by rare event, cell population, and/or contextual evaluation. In: Grohs, H.K., Husain, O.A.N. (eds.) Automated Cervical Cancer Screening, pp. 44–51. IGAKU-SHOIN Medical Publishers, New York (1994)
6. Mehnert, A.J.H.: Image analysis for the study of chromatic distribution in cell nuclei with application to cervical cancer screening. School of Information Technology and Electrical Engineering, vol. Phd. The University of Queensland, Australia (2003)
7. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. John Wiley & Sons, New York (2001)
8. Alpaydin, E.: Introduction to machine learning. The MIT Press, London (2004)
9. Fukunaga, K.: Introduction to statistical pattern recognition. Academic Press (1990)
10. Fisher, R.A.: The use of multiple measurements in taxonomic problems. Ann. Eugenics, 179–188 (1936)
11. Duda, R.O., Hart, P.E.: Pattern classification and scene analysis. John Wiley & Sons (1973)

Improvement of Audio Feature Extraction Techniques in Traditional Indian Musical Instrument

Kohshelan and Noorhaniza Wahid

Faculty of Computer Science and Information Technology
Universiti Tun Hussein Onn Malaysia
86400 Parit Raja, Batu Pahat, Johor, Malaysia
shivanuthm@yahoo.com.sg,
nhaniza@uthm.edu.my

Abstract. Traditional Indian musical instrument is one of the oldest musical instruments in the world. The musical instruments have their own importance in the field of music. Traditional Indian musical instrument could be categorized into three types such as stringed instruments, percussion instruments and wind-blown instruments. However, this paper will focus on string instruments because its show fluctuating behavior due to noise. Therefore, three techniques are selected based on the frequently used by previous researches which show some shortcoming while extracting noisy signal. The three techniques are Mel-Frequency Cepstral Coefficient (MFCC), Linear Predictive Coding (LPC) and Zero-Crossing Rate (ZCR). Hence, this research attempts to improve the feature extracting techniques by integrating Zero Forcing Equalizer (ZFE) with those extraction techniques. Three classifiers that are k-Nearest Neighbor (kNN), Bayesian Network (BNs) and Support Vector Machine (SVM) are used to evaluate the performance of audio classification accuracy. The proposed technique shows better classification accuracy when dealing with noisy signal.

Keywords: Traditional Indian music, Zero Forcing Equalizer, Feature Extraction & Classification.

1 Introduction

Traditional Indian Musical Instrument is one of the oldest musical instruments in the world. This musical instrument has their own importance in the field of music. Not much research has been done on the computational aspect of Indian music [7]. Traditional Indian musical instrument could be categorized into three types such as stringed instruments, percussion instruments and wind-blown instruments. This paper will be focusing on string instruments only due to the problems occur where its show fluctuating behaviour during different experimental setups [7]. This is due to the vibration of string instruments which produce noise in highest amplitude [5] [25] [26]. Hence, three audio feature extraction techniques are selected that are Mel-Frequency Cepstral Coefficient (MFCC), Linear Predictive Coding (LPC) and Zero-Crossing Rate (ZCR) based on the frequently used by previous researches. Nevertheless, these

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_48, © Springer International Publishing Switzerland 2014

techniques had shown some shortcoming on extracting noisy signal. [2][28]. MFCC is one of the techniques commonly used in digital signal processing [1]. The drawback of MFCC technique is it provides less accuracy if the audio signal used is noisy [2]. Similar to MFCC, Linear Predictive Coding (LPC) is another technique which offers a powerful, yet simple method to extract audio information [3]. However, the drawback of LPC is in extracting noisy signal at high amplitude due to its linear computation nature [29]. It does not take nonlinear audio signal into account. On the other hand, ZCR is useful for musical instruments measurement and endpoint detection [27]. Unfortunately, the drawback of ZCR is it measures the noisiness of the signal while it is not actually used for noisy signal [28]. In addition, the detection of end point is based on the environment, the signal been uttered [27].

In this paper, Zero Forcing Equalizer (ZFE) is proposed to be integrated with the three feature extraction techniques namely MFCC-ZFE, LPC-ZFE and ZCR-ZFE in order to improve the performance of audio classification rate. The function of ZFE is to remove the high amplitude noise in the signal. A good audio extraction technique will lead to higher accuracy of classification. Audio classification was performed in the research to classify each music clip to different class. The research evaluated the results based on the classification accuracy.

This paper will describe the overview of audio feature extraction techniques used in the next section. Section 3 will describe the experiments done in each phases to extract and classify the audio files. Meanwhile, section 4 will evaluate the original and proposed techniques of audio feature extraction and its effects on the classification result.

2 Overview of Audio Feature Extraction Techniques

Audio feature extraction is an importance process that involves transforming audio data from a high to low dimensional representation. Many audio features such as fundamental frequency, RMS (Root Mean Square) amplitude and spectral centroid are referred as a scalar features. One of these features have perceptual correlates such as pitch, loudness and brightness, although in many cases there is no standard mapping between the measured and perceived features [8]. The field of musical instruments feature extraction is a wide research area, for improving feature extraction will most likely have the major impact on the performance of an instrument classification system [9].

2.1 Mel Frequency Cepstral Coefficients (MFCCs)

Mel Frequency Cepstral Coefficients (MFCCs) are cepstral coefficients used for representing audio in a way that mimics the physiological properties of the human auditory system [10] [11]. MFCCs are commonly used in audio recognition and are finding increased use in music information recognition and genre classification systems.

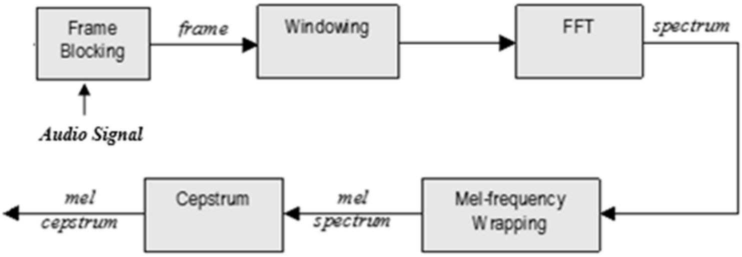


Fig. 1. MFCC Block diagram

From the MFCC block diagram shown in Fig. 1, signal from musical instrument will continuously pass into the frame blocking. Frame blocking will block the signal into frames of N samples, with adjacent frames being separated by M (M<N). The first frame consists of first N samples; second frame begins with M samples after the first frame, and overlaps it by N-M samples and so on [12]. The next step is to window each individual frame so as to minimize the signal discontinuities at the beginning and end of each frame by taper the signal to zero. After that, the signal will be processed using Fast Fourier Transform (FFT). FFT is known as fast algorithm to implement the Discrete Fourier Transform (DFT) [13]. FFT is used to speed up the processing of audio signal [29]. It will convert each frame of *N* samples from the time domain into the frequency domain.

In order to capture the important characteristics of audio, signal of the audio is expressed in the Mel frequency scale, which is a linear frequency spacing about 1000Hz. The signal was wrapped through a process named Mel-frequency wrapping. In the final step, the log mel spectrum has to be converted back to time and the result is called the MFCCs as shown in (1). The mel spectrum coefficients (and so their logarithm) are real numbers, thus it can be converted to the time domain using the Discrete Cosine Transform (DCT). Therefore,

$$C_i(n) = \sum_{m=1}^{M} S(m)\cos\left[\frac{\pi n(m-0.5)}{M}\right], \quad 0 \le m < M \tag{1}$$

where *n* is the number of MFCC, *C*$_i$*(n)* is the *n*-th MFCC coefficients of the *i*-th frame, *S(m)* is the logarithmic power spectrum of the audio signal, and M is the number of triangular filters [14].

2.2 Linear Predictive Coding (LPC)

Similar to MFCC, Linear predictive coding (LPC) is another technique which offers a powerful, yet simple method to extract audio information as shown in (2) [15]. Basically, the LPC algorithm produces a vector of coefficients that represent a smooth spectral envelope of a temporal input signal [3]. Using LPC alone was not very successful because the all pole assumption of the vocal cord transfer function was not accurate [16]. The advantage of LPC is it has high rate of audio compression [17], take short training time [16] and could remove the redundancy in signal [17]. However, LPC could not extract noisy signal at high amplitude due to its linear computation nature [5] and take long time for extracting the features [16].

$$H(z) = \frac{G}{1 + \sum_{k=1}^{p} a_p(k) z^{-k}} \quad (2)$$

,

where p is the number of poles, G is the filter gain, and $\{a_p\ (k)\}$ are the parameters that determine the value being used.

2.3 Zero-Crossing Rate (ZCR)

Zero-crossing rate (ZCR) is the rate at which the signal changes from positive to negative or otherwise. The rate at which zero crossing occurs is a simple measure of the frequency content of a signal. It is a measure of number of times in a given frame that the amplitude of the audio signals passes through a value of zero as shown in (3) [18]. In order to use ZCR to distinguish unvoiced sounds from noise and environment, the waveform can be shifted before computing the ZCR. This is particular useful if the noise is small. In addition, ZCR has high signal frequency rate and is much lower for voiced speech as compared to unvoiced speech [19]. ZCR is an important parameter for voiced/unvoiced classification and for endpoint detection [27]. The disadvantage of ZCR is it did not take noise into account [28].

$$\mathrm{ZCR} = \frac{1}{T-1} \sum_{t=1}^{T-1} \mathbb{II}\{s_t s_t - 1 < 0\} \quad (3)$$

,

where, s is a signal of length T and the indicator function $\mathbb{II}$ {A} is equal to 1 when A is true and is equal to 0 otherwise.

2.4 Zero Forcing Equalizer (ZFE)

Zero Forcing Equalizer (ZFE) is a linear receiver used in communication systems [6]. This equalizer inverts the frequency response of the channel to the received signal so as to restore the signal before the channel. This receiver is called Zero Forcing as it brings down the ISI (Inter-Symbol Interference) to zero. The meaning of ISI is when two waves added, it will create noise at high amplitude (louder sound) [6]. Therefore, ZFE is known to boost the channel noise [20]. In previous research, ZFE had been used to solve the problem of transmit antenna selection in Multiple Input Multiple Output (MIMO) systems [21]. In this research, ZFE is used to bring down the high amplitude noise to overcome the problem of audio feature extraction techniques. Fig. 2 shows the filter process of ZFE.

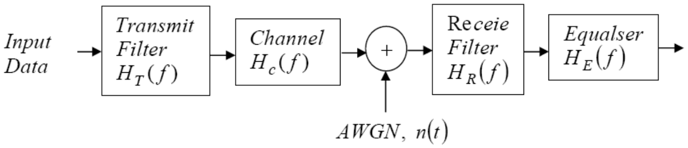


Fig. 2. Zero forcing equalizer [22]

Besides ZFE, other equalizer such as MMSE (Minimum Mean-Square Error) does not assume any stochastic mechanism of the desired and observed signals. It only makes assumptions about the noise. For example, the noise is additive zero-mean, time-independent, bounded, and known variance [23].

2.4.1 The Proposed Technique of MFCC with ZFE (MFCC-ZFE)

The integrating of ZFE with MFCC is implemented in the Frame Blocking. This is due to the particular part that boosts the high amplitude of signal [29]. Therefore, implementation of ZFE in that particular part will bring down the noise in high amplitude. Fig. 3 shows MFCC-ZFE block diagram.

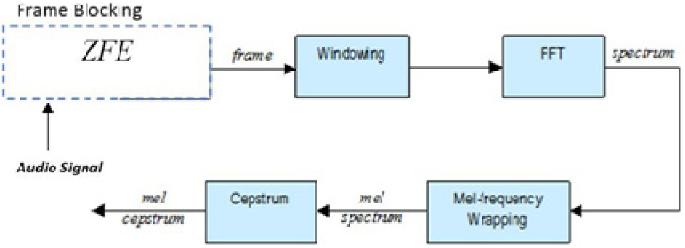


Fig. 3. MFCC-ZFE Block diagram

2.4.2 The Proposed Technique of LPC with ZFE (LPC-ZFE)

The integration process of LPC with ZFE take place in LPC Synthesizer because the process part involved the scaling of output signal in order to match the level of the input signal [30]. Hence, when the synthesizer handles the noise, the quality of the signal remains. Fig. 4 shows LPC-ZFE block diagram.

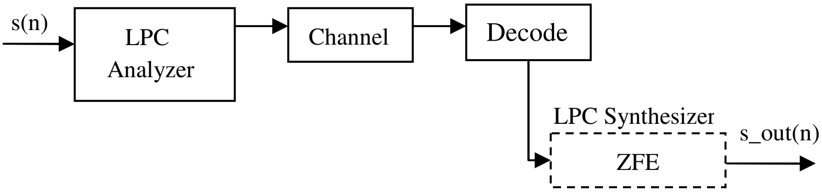


Fig. 4. LPC-ZFE Block diagram

2.4.3 The Proposed Technique of ZCR with ZFE (ZCR-ZFE)

Fig. 5 shows ZCR-ZFE block diagram where ZFE had been integrated in ZCR. ZFE was implemented in energy calculation. This is due to the occurrence of noise in high level of energy (amplitude) [18] in that particular part. Therefore, when noise in high amplitude was identified, it will be removed.

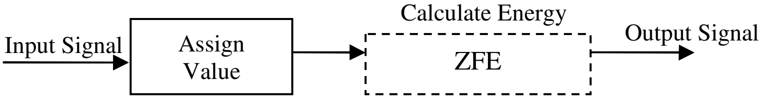


Fig. 5. ZCR-ZFE Block diagram

3 Experiments Setup

Systematic techniques needed in each phase of development so that the features extract successfully and achieve the objectives. Therefore, the proposed feature extraction programs were coded using Matlab to extract the audio features from the audio file. Weka [24] tool was used for audio classification purposes.

3.1 Audio Acquisition

The importance part in this phase is to know the characteristic of the audio. Audio file used is Veena from Traditional India Musical Instrument. There are five classes of instruments belong to Veena such as Chitra Veena, Mohan Veena, Rudra Veena, Saraswati Veena, and Vichitra Veena. Each instrument contributed to 100 audio data and was played by five musicians. Overall, there are total of 500 audio files. Table 1 shows the properties of audio file.

Table 1. Audio File Properties

Audio File Properties	Information
File type	.wav
Length/ Duration	5 seconds
Sampling Rate	16000Hz
Bit depth	16 bit (Stereo)

In the pre-processing stage, the first step is plotting the input signal. Here, only the high amplitude of the signal that considers noisy parts of audio signals were cropped. Later, the segment of audio file (signal) was transformed into linear power spectrum. The aim of this step is to apply logarithmic power spectrum known as linear spectrum. Afterward, Mel-spaced filter bank was applied to audio signals. The reason is that the Mel scale tells us exactly how to space our filter banks since this step illustrate the level of energy exists in signal.

3.2 Audio Feature Extraction

Audio feature extraction is an important process. Three audio extraction techniques such as MFCC, LPC and ZCR were implemented to extract the audio features from Traditional Indian Musical Instruments. 12 coefficient features were extracted from MFCC, 4 features from LPC and 1 feature contributed by ZCR. The value of the extracted features were averaged and fed into the respective classifiers.

3.3 Audio Classification

Three different classifiers were used, that is K-Nearest Neighbor (kNN), Bayesian Network (BNs) and Support Vector Machine (SVM) based on their good performance in audio classification [31]. The extracted features are fitted into Weka Tool [24] for classification and analyze the result. The tool generates train and test data using 10-fold cross-validation. Moreover, a comparison is performed to show the performance of the proposed techniques.

4 Analysis and Results

The experiment involves extracting audio features from MFCC, LPC, ZCR, MFCC-ZFE, LPC-ZFE and ZCR-ZFE.

Table 2. A comparison of the audio classification accuracy between the original and the proposed techniques in three different classifiers

	k-Nearest Neighbor (kNN)	Bayesian Network (BNs)	Support Vector Machine (SVM)
MFCC	94.2%	79.6%	72.6%
LPC	26.2%	35.2%	35.8%
ZCR	26.4%	31.4%	27.2%
MFCC-ZFE	98.2%	81.2%	81.4%
LPC-ZFE	31.8%	39.8%	38.8%
ZCR-ZFE	31.4%	35.8%	33.8%

In Table 2, it is clearly shows that, the proposed technique of MFCC-ZFE provides better classification rate with 98.2%, 81.2% and 81.4% when classified by KNN, BNs and SVM respectively. Obviously, the results show that MFCC-ZFE improved the classification performance by 4% when compared to original MFCC. Interestingly, SVM contributes to the biggest difference with 8.8%.

It also noted that the proposed technique of LPC-ZFE has shown a significant improvement as compared to the original technique. Specifically, by using kNN classifier, LPC-ZFE has shown an increment in classification rate with a difference of 5.6% as compared to the original LPC. Meanwhile, the proposed technique of

ZCR-ZFE has shown some improvement when compared to the original ZCR. The improvement is due to the used of SVM classifier which provides a great improvement with a difference of 6.6% in classification rate. Overall, BNs classifier has shown a better accuracy result as compared to kNN and SVM classifier.

5 Conclusion

This paper has proposed a ZFE to be integrated with MFCC, LPC and ZCR in order to improve the audio extraction technique. The techniques are called MFCC-ZFE, LPC-ZFE and ZCR-ZFE. The proposed techniques have shown some improvement in terms of providing better features which can be seen from the performance of the classification accuracy. In addition, by extracting the audio signal with the improved techniques of MFCC-ZFE, LPC-ZFE and ZCR-ZFE, this paper reveals that the ZFE is capable to bringing down the noise in high amplitude of the signal. Also, the experimental results are encouraging, illustrating that both the combination strategies lead to more accurate result. The future work will emphasize on the audio classification by using Swarm Intelligence classifier such as Simplified Swarm Optimization (SSO) [32] and Artificial Bee Colony (ABC).

Acknowledgment. This work was supported by the Office for Research, Innovation, Commercialization and Consultancy Management (ORICC) of University Tun Hussein Onn Malaysia, Johor. Grant funded by the Graduate Incentive Scheme Vot 1259.

References

1. Weeks, M.: Digital Signal Processing Using MATLAB and Wavelets, p. xi (2010), http://books.google.com.my/books?id=du6S5-4znlAC&printsec=frontcover
2. Anusuya, M.A., Katti, S.K.: Comparison of Different Speech Feature Extraction Techniques with and without Wavelet Transform to Kannada Speech Recognition. International Journal of Computer Applications 26(4), 19–24 (2011)
3. Elminir, H.K., ElSoud, M.A., El-Maged, L.A.: Evaluation of Different Feature Extraction Techniques for Continuous Speech Recognition. International Journal of Science and Technology 2(10) (2012)
4. Rajagopal, K.: Engineering Physics. PHI Learning Pvt. Ltd. (2007), http://books.google.com.my/books?id=vdXHcub8fRAC&printsec=frontcover
5. Dave, N.: Feature Extraction Methods LPC, PLP and MFCC in Speech Recognition, ijaret.org/Feature Extraction Methods LPC, PLP and MFCC.pdf (2013)
6. Kaur, N., Kansal, L.: Performance Comparison of MIMO Systems over AWGN and Rician Channels with Zero Forcing Receivers. International Journal of Wireless & Mobile Networks 5(1), 73–84 (2013)
7. Gunasekaran, S., Revathy, K.: Fractal dimension analysis of audio signals for Indian musical instrument recognition. In: ICALIP, ISBN: 978-1-4244-1723-0

8. Bullock, J.: Implementing audio feature extraction in live electronic music. PhD thesis, University of Birmingham (2008)
9. Gunasekaran, S., Revathy, K.: Recognition of Indian Musical Instruments with Multi-Classifier Fusion. In: International Conference on Computer and Electrical Engineering, Phuket (2008) ISBN: 978-0-7695-3504-3
10. Kumari, M., Kumar, P., Solanki, S.S.: Classification of North Indian Musical Instruments using Spectral Features. GESJ: Computer Science and Telecommunications 6(29) (2010)
11. Sukor, A. S.: Speaker identification using MFCC procedure and noise reduction method. Universiti Tun Hussein Onn Malaysia: Master's Project Report (2012)
12. Gadade, M.H., Jadhav, M.M.R., Deogirkar, M.S.V.: Speech Identification and Recognition Using Data Mining. Signal, 1, 0 (2010)
13. Panda, A.K., Sahoo, A.K.: Study of Speaker Recognition Systems (Doctoral dissertation) (2011)
14. Xie, C., Cao, X., He, L.: Algorithm of Abnormal Audio Recognition Based on Improved MFCC. Procedia Engineering 29, 731–737 (2012)
15. Aviv, A., Grichman, K.: Long-term prediction (2011), http://health.tau.ac.il/Communication%20Disorders/noam/noam_audio/adit_kfir/html/lpc3.htm
16. Wolf, M., Nadeu, C.: Evaluation of different feature extraction methods for speech recognition in car environment. In: Systems, Signals and Image Processing, IWSSIP 2008, pp. 359–362 (2008)
17. Gunjal, S.D., et al.: Advance Source Coding Techniques for Audio/Speech Signal: A Survey. Int. J. Computer Technology & Applications 3(4), 1335–1342 (2012)
18. Raju, N., Arjun, N., Manoj, S., Kabilan, K., Shivaprakaash, K.: Obedient Robot with Tamil Mother Tongue. Journal of Artificial Intelligence 6, 161–167 (2013)
19. Ngo, K.: Digital signal processing algorithms for noise reduction, dynamic range compression, and feedback cancellation in hearing aids (2011) (status: published)
20. Proakis, J., Salehi, M.: Digital Communications, 5th edn. McGraw-Hill, NY (2008)
21. Khademi, S., Chepuri, S.P., Leus, G., van der Veen, A.J.: Zero-forcing pre-equalization with transmit antenna selection in MIMO systems (2013)
22. Mobile Communication.: Equalization, Diversity and Coding Techniques (2009), http://mc.lctu.cn
23. Chen, J., Ma, T., Chen, W., Peng, Z.: Unsupervised robust recursive least-squares algorithm for impulsive noise filtering. Science China Information Sciences, 1–10 (2013)
24. WEKA.: Downloading and Installing WEKA (2010), http://www.cs.waikato.ac.nz/ml/weka/
25. Stulov, A., Kartofelev, D.: Vibration of strings with nonlinear supports. Applied Acoustics 76, 223–229 (2014)
26. Subramanian, M.: Carnatic Music and the Computer (2006), http://www.musicresearch.in/download.php?id=37&artname=article_37.zip
27. Khan, A.U., Bhaiya, L.P., Banchhor, S.K.: Hindi Speaking Person Identification Using Zero Crossing Rate. International Journal of Soft Computing & Engineering, 01–04 (2012)
28. Bormane, D.S., Dusane, M.: A Novel Techniques for Classification of Musical Instruments. Information and Knowledge Management, 1–8 (2013)
29. Chougule, S.V., Chavan, M.S.: Channel Robust MFCCs for Continuous Speech Speaker Recognition. In: Thampi, S.M., Gelbukh, A., Mukhopadhyay, J. (eds.) Advances in Signal Processing and Intelligent Recognition Systems. AISC, vol. 264, pp. 557–568. Springer, Heidelberg (2014)

30. McCree, A.: Low-Bit-Rate Speech Coding. Springer Handbook of Speech Processing 72, 97–105 (2008)
31. Li, T., Ogihara, M., Li, Q.: A comparative study on content-based music genre classification. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 282–289. ACM (2003)
32. Chung, Y.Y., Wahid, N.: A hybrid network intrusion detection system using simplified swarm optimization (SSO). Appl. Soft Comput. 12(9), 3014–3022 (2012)

Increasing Failure Recovery Probability of Tourism-Related Web Services

Hadi Saboohi[1], Amineh Amini[1], and Tutut Herawan[1,2]

[1] Department of Information System
Faculty of Computer Science and Information Technology
University of Malaya
50603 Kuala Lumpur, Malaysia
[2] AMCS Research Center, Yogyakarta, Indonesia
{hsaboohi,tutut}@um.edu.my,
amini@siswa.um.edu.my

Abstract. The reliability of tourism-related Web services are crucial. Users do not tend to check and use a service again once it is failed. Researches proved that a simple one-to-one replacement of a failed service is not dependable to recover a system from a total failure. In order to increase the probability of recovery of a failure we use a renovation approach to replace a set of services. Broadening the search area among the services in a graph of services enables us to increase the failure recovery probability. The time complexity is also considered and proved to be low at the failure time by transferring the time-consuming calculations to an offline phase prior to the execution time of the service. The approach is evaluated on a set of services including the tourism-related Web services. The probability of recovery substantially increased to more than 54% of the simulated failures.

Keywords: Tourism Web Services, Semantic Services, Failure Recovery, Subdigraph Replacement.

1 Introduction

Tourism is a popular global leisure activity which plays a big role with revenues of billion dollars. New services are being created by widespread availability of mobile phones. Users are increasingly exploiting complex tourism-related applications [1]. The requests can be as simple as a hotel booking or as complex as an arrangement of a complete itinerary of a trip. The users may encounter a failure during a use of a service such as a failure to book a room at an affordable hotel. For complex requests the failures are more probable because the whole service includes many inter-related smaller services. It is crucial that the enterprises fulfill users' goals even though the failures are inevitable.

It is straightforward to replace a failed Web service with a similar one [2–5]. The approach is illustrated in Figure 1. However, there are two obstructive elements. First, comparing two services to find out whether they are similar, i.e.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_49, © Springer International Publishing Switzerland 2014

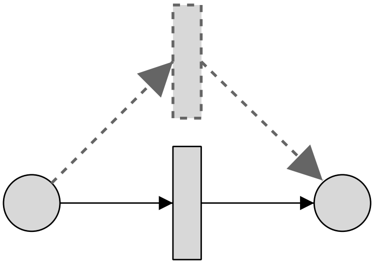


Fig. 1. Atomic Replacement (one-to-one) (Adapted from [8])

they can achieve the same results, needs a thorough search in the repository of services. Second, the chances of being a similar service available to be invoked should be high enough. The first problem is alleviated using semantics of services (known as semantic Web services) [6] in order to find a matched service. Nevertheless, even though the number of competing providers are high, the second problem is still hindering the service-based systems to offer a service with high availability. A study of failure recovery probability on publicly available semantic Web services shows that it is not highly likely to find a matched service for an assumed failing Web service [7].

Our proposal is to use a more advanced approach to increase the probability of recovery of a failed service in order to enhance the availability of tourism-related services. The enhancement is based on a renovation approach which replaces a set of services including the failed service with another similar set.

The remainder of this paper is organized as follows. Section 2 overviews major methods used to increase the probability of recovery of failed composite services. We explain our approach in Section 3 and discuss its evaluation in Section 4. The conclusion is given in Section 5.

2 Related Work

There are a number of researches carried out in the literature which investigate the recovery of composite Web services. These researches focus on replacing the failed Web service with another one as well as adapting the composite structure of the service. The aim is to hinder the whole system from a failure.

The work presented in [9] includes two algorithms which adapt the composite Web service. The algorithms first provide a backup path in order to reroute a process, and second perform a reconfiguration of a new process. The business processes are created using abstract services, i.e. the concrete services will be assigned at the execution time. A limitation is that the algorithms do not go backwards through the structure of the business process to find a new route.

The researchers in [10, 11] dealt with the problem of recovery of services from an aspect considering quality of service (QoS). The works bind the abstract services to the concrete ones taking into account the QoS values of the services.

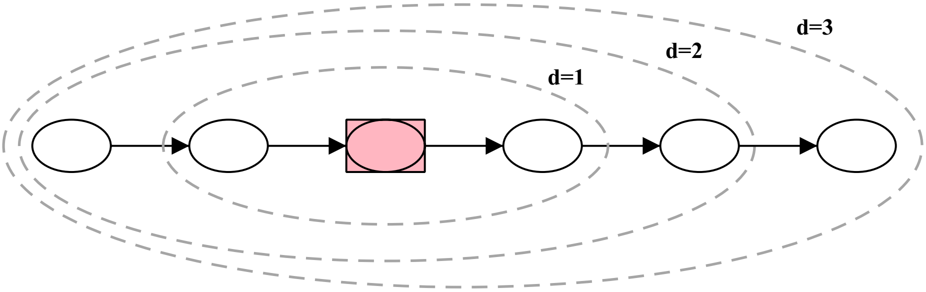


Fig. 2. Reconfiguration Region Example [12]

The main idea is to re-bind the services at run-time. The re-bind approach is performed on a so-called slice of composite service starting from the failure point (Web service). Hence, the re-binding will be performed on the non-executed set of services. It is an advantage if the remaining services which have not been executed yet are also considered to be removed from the structure, in order to find an alternative set. This may result in a change in the quantity as well as the quality of the recovered composite service's structure.

The FACTS framework in [2] used an *Alternate* exception handling strategy for composition of services. The work is categorized as an atomic replacement approach. The other approaches of recovery including composite replacement can be used to increase the failure recovery probability.

Lin et al. proposed a dynamic reconfiguration of services [12]. The approach includes several tasks: First, the faulty regions of the composite service are identified. Second, the constraints for the regions are calculated and finally, the regions are recomposed. The approach shows a good percentage of recovery of failed services since it goes backwards from the failed service to even the executed services. Figure 2 represents the region examples. The work in [13] extended the dynamic reconfiguration idea by ensuring its correctness.

There is another approach presented in [8] which uses the idea of subdigraph replacement. It modifies the composite service's structure at run-time. Transfer-

Table 1. The Most Related Researches' in Chronological Order

Paper	Idea	Binding
[9]	Backup path	Abstract
[10, 11]	Rebinding	Abstract
[2]	*Alternate* exception handling	Concrete
[12]	Region Reconfiguration	Concrete
[8]	Dynamic Substitution	Concrete

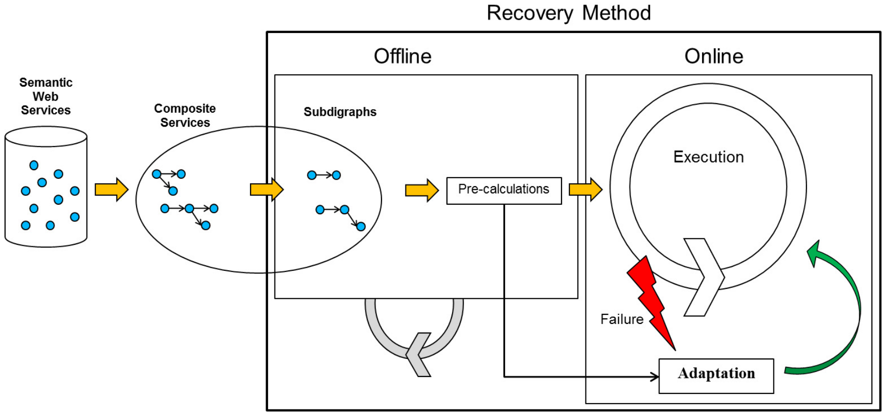


Fig. 3. Overview of the Recovery Approach [15]

ring some of the time-consuming calculations to a phase earlier, i.e. before the failure happens, makes the recovery process faster and more efficient.

There are some requirements for a failure recovery approach of composite Web services which are investigated in [14].

We have studied the failure recovery of composite services in [15] and [16]. These works are extended and evaluated for tourism-related Web services in this paper.

Table 1 represents the most related approaches. The binding methods, i.e. Abstract or Concrete, in composition structures are also identified.

3 Failure Recovery Using a Subdigraph Renovation Approach

In order to increase the probability of recovery of the composite services from a failure, we use a two-phase approach. The approach is in two offline-online phases. An overview of our proposed approach is depicted in Figure 3.

3.1 Offline Phase

The offline phase prepares the approach for the recovery. It calculates the subdigraphs of the composite services and ranks them. The ranking measure is based on Eqn. 1:

- $|Services|$: the number of services in the subdigraph.
- $t_{execution}$: the subdigraph's total execution time.
- $Cost_{execution}$: the cost (price) needed for the execution of the subdigraph.
- $|Undo|$: the number of services required to be undone if the subdigraph's execution fails.

- $Cost_{Undo}$: the cost (charge) needed for the undo (of the effects) of the subdigraph's execution.
- $\mathcal{R}$: the reliability value of the subdigraph.

$$\begin{aligned} Ranking\ Measure\ 1 = & \\ & \alpha_1(1 - |Services|) + \\ & \alpha_2 t_{execution} + \alpha_3 Cost_{execution} + \\ & \alpha_4(1 - |Undo|) + \alpha_5(1 - Cost_{Undo}) + \\ & \alpha_6(1 - \mathcal{R}) \end{aligned} \tag{1}$$

The weight values (α_1, α_2, ..., α_6) in Eqn. 1 are constrained to have a sum of 1.

Moreover, the approach looks for the subdigraphs' alternatives based on an exact match of the semantic specifications of their inputs and outputs as well as their preconditions and results (effects). Then, the alternatives are ranked to find the topmost replacement. The ranking is based on several features of the subdigraphs including functional and non-functional properties as in Eqn. 2.

$$\begin{aligned} Ranking\ Measure\ 2 = & \\ & \beta_1(1 - |Services|) + \\ & \beta_2(1 - t_{execution}) + \beta_3(1 - Cost_{execution}) + \\ & \beta_4 \mathcal{R} \end{aligned} \tag{2}$$

Similarly, the weight values (β_1, β_2, ..., β_4) in Eqn. 2 are constrained to have a sum of 1 as well.

Finally, the best "Replacement Subdigraph" for a Web service that is assumed to fail is calculated by combining two mentioned rank measures as follows in Eqn. 3. The sum of γ_1 and γ_2 equals to 1.

$$\begin{aligned} Final\ Rank = & \\ & \gamma_1(Ranking\ Measure\ 1) + \\ & \gamma_2(Ranking\ Measure\ 2) \end{aligned} \tag{3}$$

3.2 Online Phase

The online phase is triggered by a failure at run-time. The best replacement is ready from the offline phase for every constituent service of the composite. Hence, without any delay the original service is renovated and its execution continues. The adaptation is illustrated in Figure 4.

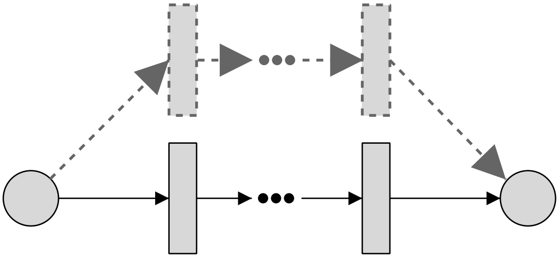


Fig. 4. Composite-to-Composite (Subdigraph) Replacement (Adapted from [8])

4 Evaluation

4.1 Test Collection

A main obstacle in evaluating failure recovery probability of composite Web services is a lack of a standard test collection. There are some test collections of semantic Web services such as SWS-TC [17], and OWLS-TC [18]; however, they are collections of atomic services and they do not include any composite service [19].

We examined the services in OWLS-TC and our findings are as follows. For 246 of the services (out of 1083) available in OWLS-TC at least an input-output (IO) match exists, which is 23% of the services in the collection. By the term IO match, we mean that the inputs and outputs of a service A are exactly using the same concepts as of the inputs and outputs of another service B respectively. In terms of IOPR (Input, Output, Precondition and Result) match, the number is 232, 21% of the test data. The existence of an exact match in OWLS-TC is shown in Figure 5 [15].

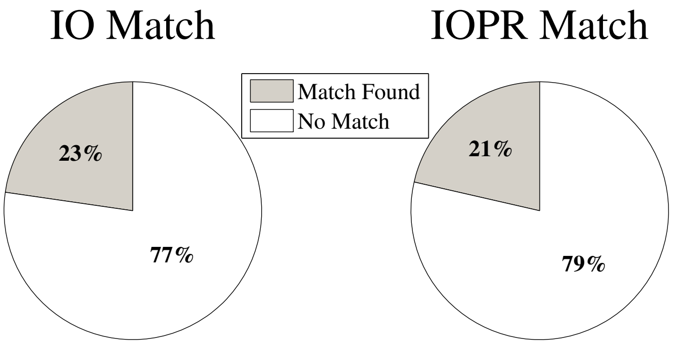


Fig. 5. Existence of an Exact Match in OWLS-TC [15]

Table 2. Services in OWLS-TC

ID	Name	Filename (.owls)	IO	PR	Class
...					
204	Recommended price of car model	caryear_recommendedpriceineuro_service	2,1	0,0	
205	Car price	car_priceauto_service	1,2	0,0	
206	T-car price	car_pricecolor_service	1,2	0,0	
207	Car Price quality	car_pricequality_service	1,2	0,0	
208	car price report	car_pricereport_service	1,2	0,0	
209	car price	car_price_service	1,1	0,0	209
210	CarRecommendedPrice	car_recommendedpriceindollar_service	1,1	0,0	
211	car Recommended price	car_recommendedpriceineuro_service	1,1	0,0	
212	CarRecommendedPrice	car_recommendedprice_service	1,1	0,0	
213	car report	car_report_service	1,1	0,0	
214	car price	car_taxedpriceprice_service	1,2	0,0	
215	Car TaxedPrice Report	car_taxedpricereport_service	1,2	0,0	
216	CarTechnology	car_technology_service	1,1	0,0	
217	Car Year Price	car_yearprice_service	1,2	0,0	
...					
235	CheckCostAndHealingPlan	CheckCostAndHealingPlan_service	2,1	0,0	
236	CheckEquipmentAvailability	CheckEquipmentAvailability_service	4,1	0,0	
237	CheckHospitalAvailability	CheckHospitalAvailability_service	3,1	0,0	237
238	CheckPersonnelAvailability	CheckPersonnelAvailability_service	3,1	0,0	237
239	CheckRoomAvailability	CheckRoomAvailability_service	3,1	0,0	237
240	citycity route finder	citycity_arrowfigure_service	2,1	1,0	
241	City2CityRouteFinderService	citycity_map_service	2,1	1,0	
242	HotelReserveService	citycountryduration__HotelReserveservice	3,1	1,1	
243	AccomodationInfoService	citycountry_accommodation_service	2,1	1,0	
244	CityCountaryInfoService	citycountry_destinationhotel_service	2,2	1,0	
245	HotelInfoService	citycountry_hotel_service	2,1	1,0	245
...					
594	leynthu rent a car	lenthu_rentcar_service	1,1	0,0	209
595	AvailableVideoService	linguisticexpression_videomedia_service	1,1	0,0	
596	RouteFinderService	locationlocation_arrowfigure_service	2,1	1,0	
597	LocationLocationIcon	locationlocation_icon_service	2,1	1,0	
598	RouteFinderService	locationlocation_map_service	2,1	1,0	598
599	SRI RouteFinderService	locationlocation_map_SRIservice	2,1	1,0	598
600	LocationTravelInfo	location_icon_service	1,1	0,0	
601	LocationPhotographs	location_photograph_service	1,1	0,0	
...					
881	SPP	shoppingmall_pricepurchaseableitemrange_service	1,3	0,0	
882	SHOPPINGMALLPURCHASEABLEITEMPRICE	shoppingmall_purchaseableitemprice_service	1,2	0,0	
...					

Some of 1083 services in OWLS-TC are listed in Table 2. The table identifies service ID, its name and Filename (with .owls suffix as their filename extensions), number of inputs and outputs (in IO) column, number of preconditions and results (in PR) column, and service class ID. The (service) ID is a sequential number that was assigned for the services. The service class ID is another value which was assigned for the service classes, and identifies the services with the same IO concepts. The list of all 1083 services is available in [15].

In order to show the improvement caused by the use of our model, we synthetically created a collection of composite services from atomic services available in OWLS-TC.

The generation of the services was based on the concept matching of the inputs and outputs of the services. Hence, we joined two services C and D in order to generate a composite (called CD) if the inputs of service D matches the outputs of service C. The services were specifically chosen from tourism-related services, which have some inputs and outputs from travel ontology available in OWLS-TC.

4.2 Results

We simulated the execution of the services and we assumed a failure of a service in the structure for each execution.

The applicability of a one-to-one approach to recover the services from a failure was tested. For every failure, we searched through the test collection to find a matched atomic service. Furthermore, we applied our approach to the failed composite service as well. The structure was examined in order to find a subdigraph of services to be removed and replaced by another set of services which was pre-calculated prior to the execution.

In total we created 4000 composite services with different graph orders (of equal or greater than 2 services) and various structures.

- The approach with only a one-to-one replacement shows that in 13% of the execution failures it can recover the failure by replacing a failed service with another matched service in the collection.
- The proposed approach recovered these failure cases by a one-to-one replacement approach as well. Moreover, our approach could find a set of services to be replaced by its exact match set in 41% of other cases. The cases were including one-to-many and many-to-many replacements. So, our proposed approach could recover the failure in more than 54% of the test cases.

Figure 6 illustrates the results.

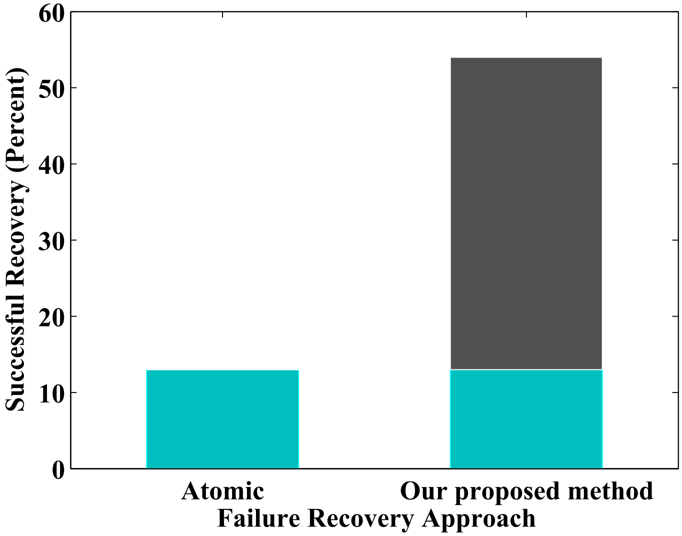


Fig. 6. The probability of recovery for a one-to-one (atomic) replacement approach and our proposed approach

5 Conclusion

Users usually discard a software which is unreliable and does not fulfill their needs. It is critical to either prevent a system from a failure or to recover it when a failure occurs. It is nearly impossible to prevent all the failures, so the methods should be able to recover a service-based system with a high probability. Tourism-related services are a major category of the services which are used more often. We evaluated our approach which prepares the execution of the services in an offline phase. It calculates a set of services to be replaced in lieu of a subdigraph of services including an assumed failing service. At the failure time our approach renovates the structure in order to survive the composite service from a total failure.

Our recovery approach for composite semantic services is evaluated on a set of travel technology, i.e. tourism-related, services. The results show a substantial improvement in recovery of more than a half of simulated failures of the composite services.

Acknowledgment. This research is supported by High Impact Research (HIR) Grant, University of Malaya No. UM.C/625/HIR/MOHE/SC/13/2 from Ministry of Higher Education.

References

1. Rodriguez-Sanchez, M., Martinez-Romo, J., Borromeo, S., Hernandez-Tamames, J.: GAT: Platform for automatic context-aware mobile services for m-tourism. Expert Systems with Applications 40, 4154–4163 (2013)
2. Liu, A., Li, Q., Huang, L., Xiao, M.: FACTS: A framework for fault-tolerant composition of transactional web services. IEEE Transactions on Services Computing 3, 46–59 (2010)
3. Angarita, R., Cardinale, Y., Rukoz, M.: FaCETa: Backward and forward recovery for execution of transactional composite ws. In: International Workshop on REsource Discovery (RED), Heraklion, Greece, pp. 89–103 (2012)
4. Subramanian, S., Thiran, P., Narendra, N.C., Mostefaoui, G.K., Maamar, Z.: On the enhancement of BPEL engines for self-healing composite web services. In: International Symposium on Applications and the Internet, pp. 33–39. IEEE Computer Society, Washington, DC (2008)
5. Taher, Y., Benslimane, D., Fauvet, M.C., Maamar, Z.: Towards an approach for web services substitution. In: 10th International Database Engineering and Applications Symposium (IDEAS), pp. 166–173 (2006)
6. McIlraith, S.A., Son, T.C., Zeng, H.: Semantic web services. IEEE Intelligent Systems 16, 46–53 (2001)
7. Saboohi, H., Abdul Kareem, S.: Increasing the failure recovery probability of atomic replacement approaches. In: Asia-Oceania Top University League on Engineering Student Conference (AOTULE), Kuala Lumpur, Malaysia, p. 123 (2012)
8. Möller, T., Schuldt, H.: OSIRIS Next: Flexible semantic failure handling for composite web service execution. In: Fourth International Conference on Semantic Computing (ICSC), Los Alamitos, CA, USA, pp. 212–217 (2010)
9. Yu, T., Lin, K.J.: Adaptive algorithms for finding replacement services in autonomic distributed business processes. In: The 7th International Symposium on Autonomous Decentralized Systems (ISADS), pp. 427–434 (2005)
10. Canfora, G., Di Penta, M., Esposito, R., Villani, M.L.: QoS-aware replanning of composite web services. In: IEEE International Conference on Web Services (ICWS), pp. 121–129 (2005)
11. Canfora, G., Di Penta, M., Esposito, R., Villani, M.L.: A framework for QoS-aware binding and re-binding of composite web services. Journal of Systems and Software 81, 1754–1769 (2008)
12. Lin, K.J., Zhang, J., Zhai, Y., Xu, B.: The design and implementation of service process reconfiguration with end-to-end QoS constraints in SOA. Service Oriented Computing and Applications (SOCA) 4, 157–168 (2010)
13. Li, Y., Zhang, X., Yin, Y., Lu, Y.: Towards functional dynamic reconfiguration for service-based applications. In: IEEE World Congress on Services (SERVICES), pp. 467–473 (2011)
14. Saboohi, H., Abdul Kareem, S.: Requirements of a recovery solution for failure of composite web services. International Journal of Web & Semantic Technology (IJWeST) 3, 15–21 (2012)
15. Saboohi, H.: An Automatic Failure Recovery Method for World-altering Composite Semantic Web Services. PhD thesis, University of Malaya (2013)
16. Saboohi, H., Abdul Kareem, S.: An automatic subdigraph renovation plan for failure recovery of composite semantic web services. Frontiers of Computer Science 7, 894–913 (2013)

17. Ganjisaffar, Y., Saboohi, H.: Semantic web services' test collection SWS-TC (2006), http://www.semwebcentral.org/projects/sws-tc/
18. OWLS-TC: OWL-S service retrieval test collection (2010), http://www.semwebcentral.org/projects/owls-tc/
19. Saboohi, H., Abdul Kareem, S.: A resemblance study of test collections for world-altering semantic web services. In: International Conference on Internet Computing and Web Services (ICICWS) in The International MultiConference of Engineers and Computer Scientists (IMECS), vol. I, pp. 716–720. International Association of Engineers, Newswood Limited, Hong Kong (2011)

Mining Critical Least Association Rule from Oral Cancer Dataset

Zailani Abdullah[1], Fatiha Mohd[1], Md Yazid Mohd Saman[1],
Mustafa Mat Deris[2], Tutut Herawan[3], and Abd Razak Hamdan[4]

[1] School of Informatics & Applied Mathematics, Universiti Malaysia Terengganu
[2] Faculty of Science Computer and Information Technology,
Universiti Tun Hussein Onn Malaysia
[3] Faculty of Computer Science & Information Technology, University of Malaya
[4] Faculty of Information Science &Technology, Universiti Kebangsaan Malaysia
{zailania,yazid}@umt.edu.my, mpfatihah@yahoo.com,
mmustafa@uthm.edu.my, tutut@um.edu.my, arh@ftsm.ukm.my

Abstract. Data mining has attracted many research attentions in the information industry. One of the important and interesting areas in data mining is mining infrequent or least association rule. Typically, least association rule is referred to the infrequent or uncommonness relationship among a set of item (itemset) in database. However, finding this rule is more difficult than frequent rule because they may contain only fewer data and thus require more specific measure. Therefore, in this paper we applied our novel measure called Critical Relative Support (CRS) to mine the critical least association rule from the medical dataset called Oral-Cancer-HUSM-S1. The result shows that CRS can be use to determine the least association rule and thus proven its scalability.

Keywords: Critical, least association rules, medical dataset.

1 Introduction

Association rules mining (ARM) is among the very famous topics in data mining. It aims at searching previously unknown, interesting and useful patterns from transactional databases. Typically, each transaction in database contains a set of items (a.k.a itemset). The problem of ARM was first introduced by Agrawal [1] and mainly focused on analyzing market-basket transactions in term of customer buying patterns. Until this recent, it has been successfully applied in numerous domain applications such as banking, marketing, telecommunication, medicine, etc. Association rule is an implication expression of A → B (such as Milk and Diaper → Coke), where A and B are itemsets. An item is said to be frequent if it appears more than a minimum support threshold. Besides that, minimum confidence threshold is always used to present the strength or reliability of association rule.

Least item is a contradicted definition of frequent item [2]. It is an itemset whose rarely or uncommonly found in the database. In certain situations, least association rule is very important especially in discovering something that is rarely occurring but

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_50, © Springer International Publishing Switzerland 2014

very interesting, such as in the field of educational data mining [3,4,5], text mining [6,7], information visualization [8,9,10], medical diagnosis [11,12] and etc. Generally, many series of ARM algorithms are depend on the minimum supports-confidence framework. For that reason, lowering the value of minimum support can assist in capturing the least items from the database. The problem is, it will indirectly force to produce abundant numbers of undesired rules. As a result, the process of determining which rules that are really interesting becomes more tedious and complicated. Furthermore, the lowering the minimum support will also proportionally intensify the memory consumption and its complexity. Therefore, due to the difficulties in the algorithms [13] development, measures formulation and it may require excessive of computational cost, there are very limited efforts have been put forward.

This paper attempts to examine and finally overcome these drawbacks by highlighting three foremost contributions. First, a novel measure called Critical Relative Support (CRS) [14] is employed to determine the most interesting least association rule. This rule is also known as critical least association rule. A range of CRS is always in 0 and 1. The more CRS value reaches to 1, the more interestingness of that particular rule. Second, SLP-Growth [15] algorithm and trie data structure called SLP-Tree are applied. In order to ensure that only certain least items are captured, Interval Least Support (ILSupp) is used in the SLP-Growth algorithm. Third, Oral-Cancer-HUSM-S1 dataset has been used in the experiment with CRS measure. Resulting from the experiments is very important to evaluate the scalability of CRS.

The reminder of this paper is organized as follows. Section 2 describes the related work. Section 3 explains the proposed method. This is followed by performance analysis thorough the experiment test in section 4. Finally, conclusion and future direction are reported in section 5.

2 Related Work

In certain domain applications, least association rule is very interesting as compared to the common rule. Thus, several works have been put forward using variety of algorithms and measures.

Hoque *et al.* [16] proposed Frequent Rare Itemset Mining Algorithm (FRIMA) to generate all the rare itemset. FRIMA uses bottom-up approach to find both the frequent and rare itemsets based on the downward closure property of Apriori. However, still it suffers from the combinatorial explosion and highly in memory consumption. Tsang *et al.* [17] introduced RP-Tree Algorithm to mine the least association rules from tree data structure and employed the information gain measure. However, construction of RP-Tree requires several conditions and filtration. Zhou *et al.* [7] suggested an approach to mine the association rule by considering only least itemset. The limitation is, Matrix-based Scheme (MBS) and Hash-based scheme (HBS) algorithms are facing the expensive cost of hash collision. Ding [18] proposed Transactional Co-occurrence Matrix (TCOM for mining association rule among rare items. However, the implementation of this algorithm is too costly. Yun *et al.* [9]

proposed the Relative Support Apriori Algorithm (RSAA) to generate least itemsets. The challenge is if the minimum allowable relative support is set close to zero, it takes similar time taken as performed by Apriori. Koh *et al.* [8] introduced Apriori-Inverse algorithm to mine least itemsets without generating any frequent rules. The main constraints are it suffers from too many candidate generations and time consumptions during generating the rare ARs. Liu *et al.* [19] proposed Multiple Support Apriori (MSApriori) algorithm to extract the least rules. In actual implementation, this algorithm is still suffered from the "rare item problem". Most of the proposed approaches [7,8,9,18,19]using the percentage-based approach in order to improve the performance of existing single minimum support based approaches.

Brin *et al.* [20] presented objective measures called lift and chi-square to measure the correlation of association rule. Lift compares the frequency of pattern against a baseline frequency computed under statistical independence assumption. Instead of lift, there are quite a number interesting measures have been proposed for association rules. Omiecinski [21] introduces two interesting measures based on downward closure property called all confidence and bond. Lee *et al.* [22] proposes two algorithms for mining all confidence and bond correlation patterns by extending the frequent pattern-growth methodology. Han *et al.* [23] proposed FP-Growth algorithm which break the two bottlenecks of Apriori series algorithms. Currently, FP-Growth is one of the fastest approach and most popular algorithms for frequent itemsets mining. This algorithm is based on a prefix tree representation of database transactions (called FP-tree).

3 Proposed Method

In this section, the employed measure and the algorithm are discussed in details.

3.1 Critical Relative Support (CRS)

Throughout this section the set $I = \{i_1, i_2, \cdots, i_{|A|}\}$, for $|A| > 0$ refers to the set of literals called set of items and the set $D = \{t_1, t_2, \cdots, t_{|U|}\}$, for $|U| > 0$ refers to the data set of transactions, where each transaction $t \in D$ is a list of distinct items $t = \{i_1, i_2, \cdots, i_{|M|}\}$, $1 \le |M| \le |A|$ and each transaction can be identified by a distinct identifier TID.

Definition

Definition 1. (Least Items). *An itemset X is called least item if* $\alpha \le \text{supp}(X) \le \beta$, *where* α *and* β *is the lowest and highest support, respectively.*

The set of least item will be denoted as Least Items and

$$\text{Least Items} = \{X \subset I \mid \alpha \le \text{supp}(X) \le \beta\}$$

Definition 2. (Frequent Items). *An itemset X is called frequent item if* $\operatorname{supp}(X) > \beta$, *where* β *is the highest support.*

The set of frequent item will be denoted as Frequent Items and

$$\text{Frequent Items} = \{X \subset I \mid \operatorname{supp}(X) > \beta\}$$

Definition 3. (Merge Least and Frequent Items). *An itemset X is called least frequent items if* $\operatorname{supp}(X) \geq \alpha$, *where* α *is the lowest support.*

The set of merging least and frequent item will be denoted as LeastFrequent Items and

$$\text{LeastFrequent Items} = \{X \subset I \mid \operatorname{supp}(X) \geq \alpha\}$$

LeastFrequent Items will be sorted in descending order and it is denoted as

$$\text{LeastFrequent Items}^{\text{desc}} = \left\{ \begin{array}{l} X_i \middle| \operatorname{supp}(X_i) \geq \operatorname{supp}(X_j),\ 1 \leq i, j \leq k,\ i \neq j, \\ k = |\text{LeastFrequent Items}|,\ x_i, x_j \subset \text{LeastFrequent Items} \end{array} \right\}$$

Definition 4. (Ordered Items Transaction). *An ordered items transaction is a transaction which the items are sorted in descending order of its support and denoted as* t_i^{desc}, *where*

$$t_i^{\text{desc}} = \text{LeastFrequentItems}^{\text{desc}} \cap t_i, 1 \leq i \leq n, \left|t_i^{least}\right| > 0, \left|t_i^{frequent}\right| > 0 .$$

An ordered items transaction will be used in constructing the proposed model, so-called LP-Tree.

Definition 5. (Significant Least Data). *Significant least data is one which its occurrence less than the standard minimum support but appears together in high proportion with the certain data.*

Definition 6. (Critical Relative Support). *A Critical Relative Support (CRS) is a formulation of maximizing relative frequency between itemset and their Jaccard similarity coefficient.*

The value of Critical Relative Support denoted as CRS and

$$\operatorname{CRS}(I) = \max\left(\left(\frac{\operatorname{supp}(\mathrm{A})}{\operatorname{supp}(\mathrm{B})}\right), \left(\frac{\operatorname{supp}(\mathrm{B})}{\operatorname{supp}(\mathrm{A})}\right)\right) \times \left(\frac{\operatorname{supp}(A \Rightarrow B)}{\operatorname{supp}(\mathrm{A}) + \operatorname{supp}(\mathrm{B}) - \operatorname{supp}(A \Rightarrow B)}\right)$$

CRS value falls between 0 and 1, and determines by multiplying the highest value either supports of antecedent divide by consequence or in another way around with their Jaccard similarity coefficient. It is a measurement to show the level of CRS between the combination of Least Items and Frequent Items either as antecedent or consequence, respectively.

3.2 Algorithm Development

Determine Interval Support for Least Itemset

Let I is a non-empty set such that $I = \{i_1, i_2, \cdots, i_n\}$, and D is a database of transactions where each T is a set of items such that $T \subset I$. An item is a set of items. A k-itemset is an itemset that contains k items. An itemset is said to be least if the support count satisfies in a range of threshold values called Interval Support (ISupp). The Interval Support is a form of ISupp (ISMin, ISMax) where ISMin is a minimum and ISMax is a maximum values respectively, such that $\text{ISMin} \geq \phi$, $\text{ISMax} > \phi$ and $\text{ISMin} \leq \text{ISMax}$. The set is denoted as R_k. Itemsets are said to be significant least if they satisfy two conditions. First, support counts for all items in the itemset must greater ISMin. Second, those itemset must contain at least one of the least items. In brevity, the significant least itemset is a union between the least items and the frequent items, and also the existence of intersection between them.

Construct Significant Least Pattern Tree

A Significant Least Pattern Tree (SLP-Tree) is a compressed representation of significant least itemsets. This trie data structure is constructed by scanning the dataset of single transaction at a time and then mapping onto path in the SLP-Tree. In the SLP-Tree construction, the algorithm constructs a SLP-Tree from the database. The SLP-Tree is built only with the items that satisfy the ISupp. In the first step, the algorithm scans all transactions to determine a list of least items, LItems and frequent items, FItems (least frequent item, LFItems). In the second step, all transactions are sorted in descending order and mapping against the LFItems. It is a must in the transactions to consist at least one of the least items. Otherwise, the transactions are disregard. In the final step, a transaction is transformed into a new path or mapped into the existing path. This final step is continuing until end of the transactions. The problem of existing FP-Tree are it may not fit into the memory and expensive to build. FP-Tree must be built completely from the entire transactions before calculating the support of each item. Therefore, SLP-Tree is an alternative and more practical to overcome these limitations.

Generate Significant Least Pattern Growth (SLP-Growth)

SLP-Growth is an algorithm that generates significant least itemsets from the SLP-Tree by exploring the tree based on a bottom-up strategy. ‘Divide and conquer’ method is used to decompose task into a smaller unit for mining desired patterns in conditional databases, which can optimize the searching space. The algorithm will extract the prefix path sub-trees ending with any least item. In each of prefix path sub-tree, the algorithm will recursively execute to extract all frequent itemsets and finally built a conditional SLP-Tree. A list of least itemsets is then produced based on the suffix sequence and also sequence in which they are found. The pruning processes in SLP-Growth are faster than FP-Growth since most of the unwanted patterns are already cutting-off during constructing the SLP-Tree data structure. The complete SLP-Growth algorithm is shown in Fig. 1.

```
Read dataset, D
Set Interval Support (ISMin, ISMax)
for items, I in transaction, T do
    Determine support count, ItemSupp
end for loop
Sort ItemSupp in descending order, ItemSuppDesc
for ItemSuppDesc do
    Generate List of frequent items, FItems > ISMax
end for loop
for ItemSuppDesc do
    Generate List of least items, ISMin <= LItems < ISMax
end for loop
Construct Frequent and Least Items, FLItems = FItems U LItems
for all transactions,T do
    if (LItems ∩ I in T > 0) then
        if (Items in T = FLItems) then
            Construct items in transaction
            in descending order, TItemsDesc
        end if
    end if
end for loop
for TItemsDesc do
    Construct SLP-Tree
end for loop
for all prefix SLP-Tree do
    Construct Conditional Items, CondItems
end for loop
for all CondItems do
    Construct Conditional SLP-Tree
end for loop
for all Conditional SLP-Tree do
    Construct Association Rules, AR
end for loop
for all AR do
    Calculate Support and Confidence
    Apply Correlation
end for loop
```

Fig. 1. SLP-Growth Algorithm

4 Result and Discussion

In this section, the performance analysis against Oral Cancer Dataset is elaborated in details.

4.1 Experimental Setup

The experiment has been performed on Intel® Core™ 2 Quad CPU at 2.33GHz speed with 4GB main memory, running on Microsoft Windows Vista. All algorithms have been developed using C# as a programming language.

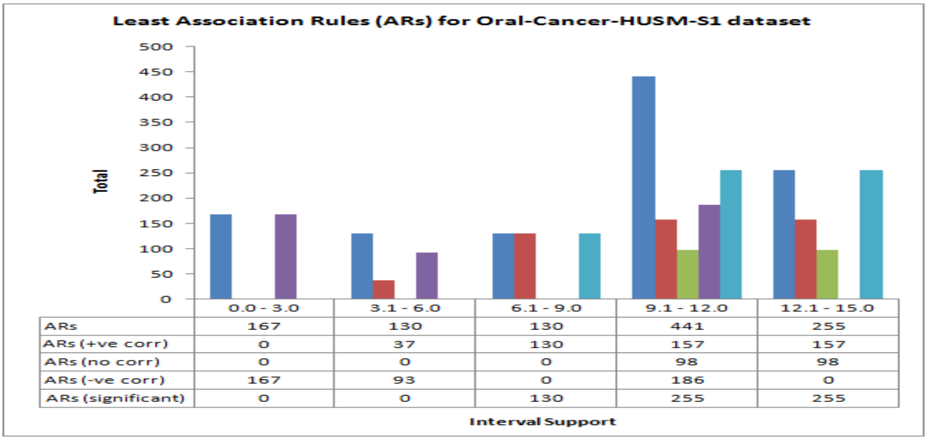
4.2 Oral Cancer Dataset

The experiment was conducted on Oral-Cancer-HUSM-S1 dataset. It contains the records of Oral Cancer of stage 1. The aim of the original dataset is to diagnose the overall stages of the oral cancer. The dataset was obtained from Otorhinolaryngology Clinic at Hospital Universiti Sains Malaysia (HUSM), Kelantan. The original data consist of 27 attributes and 82 records. After solving the problems of missing values, outliers, inconsistent and imbalance records, the final datasets consist of 210 records in which 48, 40, 64 and 57 of the records are classified as stage 1, 2, 3 and 4, respectively. The dataset is based on the records of all the patients who have been reported with a lesion and treated from January 2001 until December 2010.

In order to produce in a format of Oral-Cancer-HUSM-S1 categorical dataset, the original attributes are mapped into new attributes *id*. Only 25 attributes are selected and another 2 more attributes (Case_Id and Overall stage) are ignored. Item is constructed based on the combination of attribute id and its values. For simplicity, let consider an attribute "Age" with input value of "1". Here, an item "11" will be constructed by means of a combination of an attribute id (first character) and its input value (last character). Interval support of 0% to 15% was employed for this experiment. Support, Confidence, CRS and IS Measure (IS) are also used to quantify the association rules. In order to ensure that only the least association rule will be extracted, ISupp is set in a range of 0% to 15%. By embedding SLP-Growth algorithm with ISupp feature, only 1,123 association rules are produced. Association rules are formed by applying the relationship of an item or many items to an item (cardinality: many-to-one). At this point, the maximum number of items appears in each association rule is set to 6. Fig. 2 depicted the correlation's classification of least association rules. For this dataset, the rule is categorized as significant if it has positive correlation and CRS should be at least 0.6.

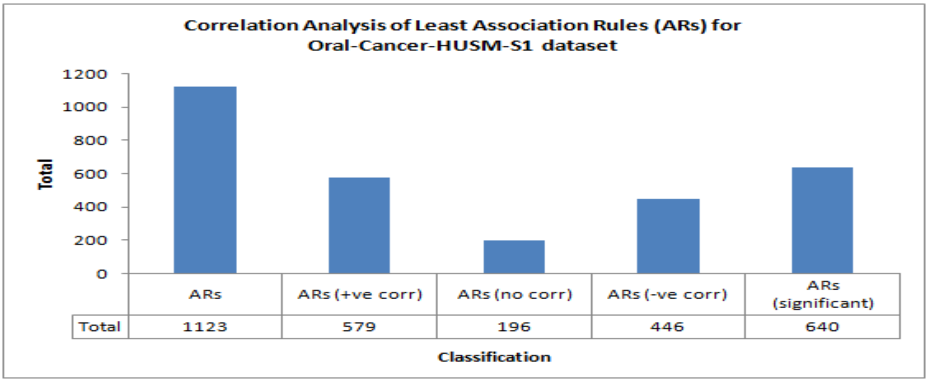
Table 1 shows some least association rules with the different measures. The support, confidence, CRS and correlation values of all selected rules are 8.33, 66.67, 0.69 and 6.40, respectively. The consequence of these association rules is 13 (Bleeding factor) and the most dominant attribute is 20 (Gender is Female). The attribute 20 appears 100% from the 130 association rules or 11.58% from overall 1,123 association rules. Therefore, based on this finding, details analysis and study by physician are recommended to discover the level of significant of these least association rules. It may reveal something interesting and great contribution in the domain knowledge of medicine. Fig. 3 illustrates the summarization of correlation analysis with different ISupp.

The result indicates that CRS successfully in producing the less number of association rule as compared to the others measures. The typical support or confidence measure alone is not a suitable measure to be employed to discover the interesting association rules. Although, the correlation measure can be used to capture the interesting association rule (by ignoring no correlation or lift is 1), the number of rules generated by CRS is still 38% lesser than lift. Therefore, CRS is proven to be more efficient and outperformed the benchmarked measures for discovering the interesting association rule from the dataset. Generally, the total numbers of association rule are kept decreased when the predefined Interval Supports thresholds are increased.



	0.0 - 3.0	3.1 - 6.0	6.1 - 9.0	9.1 - 12.0	12.1 - 15.0
ARs	167	130	130	441	255
ARs (+ve corr)	0	37	130	157	157
ARs (no corr)	0	0	0	98	98
ARs (-ve corr)	167	93	0	186	0
ARs (significant)	0	0	130	255	255

Fig. 2. Correlation analysis of interesting ARs using variety Interval Supports



	ARs	ARs (+ve corr)	ARs (no corr)	ARs (-ve corr)	ARs (significant)
Total	1123	579	196	446	640

Fig. 3. Classification of association rules using correlation analysis

Table 1. Selected of Interesting Association Rules

ARs	Support	Confidence	CRS	Corr (Lift)
20 --> 13	8.33	66.67	0.69	6.40
240 20 --> 13	8.33	66.67	0.69	6.40
250 20 --> 13	8.33	66.67	0.69	6.40
240 250 20 --> 13	8.33	66.67	0.69	6.40

5 Conclusion

Mining least association rule is undeniable a very crucial in discovering the rarity or irregularity relationship among itemset in database. It is quite complicated, computationally expensive and absolutely required special measurement in order to capture the least rules. In medical context, detecting the irregularity association rule is very useful and sometime can help save human's life. However, formulating an appropriate measure and finding a scalable mining algorithm to extract the respective rules are very challenging. Therefore, Critical Relative Support (CRS) measure and SLP-Growth algorithm are employed in the experiment of medical datasets. Here, Oral-Cancer-HUSM-S1 dataset has been used for evaluations. The result shows that CRS and SLP-Growth algorithm can be used in detecting the critical least association rules with highly correlated, and thus verify it scalabilities.

Acknowledgement. This work was supported by Research Acculturation Collaborative Effort (RACE) 2013 grant scheme, Ministry of Education.

References

[1] Han, J., Pei, H., Yin, Y.: Mining Frequent Patterns Without Candidate Generation. In: Proceeding SIGMOD 2000, Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data (SIGMOD 2000), pp. 1–12 (2000)

[2] Abdullah, Z., Herawan, T., Deris, M.M.: Detecting Definite Least Association Rule in Medical Database. In: Herawan, T., Deris, M.M., Abawajy, J. (eds.) Proceedings of the First International Conference on Advanced Data and Information Engineering (DaEng 2013). LNEE, vol. 285, pp. 127–134. Springer, Heidelberg (2014)

[3] [3] Herawan, T., Vitasari, P., Abdullah, Z.: Mining Interesting Association Rules of Student Suffering Mathematics Anxiety. In: Zain, J.M., Wan Mohd, W.M.b., El-Qawasmeh, E. (eds.) ICSECS 2011, Part II. CCIS, vol. 180, pp. 495–508. Springer, Heidelberg (2011)

[4] Herawan, T., Vitasari, P., Abdullah, Z.: Mining critical least association rules of student suffering language and social anxieties. Int. J. of Continuing Engineering Education and Life-Long Learning 23(2), 128–146 (2013)

[5] Abdullah, Z., Herawan, T., Deris, M.M.: Mining Significant Least Association Rules Using Fast SLP-Growth Algorithm. In: Kim, T.-h., Adeli, H. (eds.) AST/UCMA/ ISA/ACN 2010. LNCS, vol. 6059, pp. 324–336. Springer, Heidelberg (2010)

[6] Kiran, R.U., Reddy, P.K.: An Improved Multiple Minimum Support Based Approach to Mine Rare Association Rules. In: Proceeding of IEEE Symposium on Computational Intelligence and Data Mining, pp. 340–347 (2009)

[7] Zhou, L., Yau, S.: Assocation Rule and Quantative Association Rule Mining among Infrequent Items. In: Proceeding of ACM SIGKDD 2007, Article No. 9 (2007)

[8] Koh, Y.S., Rountree, N.: Finding Sporadic Rules using Apriori-Inverse. In: Ho, T.-B., Cheung, D., Liu, H. (eds.) PAKDD 2005. LNCS (LNAI), vol. 3518, pp. 97–106. Springer, Heidelberg (2005)

[9] Yun, H., Ha, D., Hwang, B., Ryu, K.H.: Mining Association Rules on Significant Rare Data using Relative Support. The Journal of Systems and Software 67(3), 181–191 (2003)

[10] Herawan, T., Abdullah, Z., Mohd, W.M.W., Noraziah, A.: CLAR-Viz: Critical Least Association Rules Visualization. In: The 5th International Conference on Advanced Science and Technology (AST 2013), Hiddenbay Hotel, Yeosoo, South Korea, April 26-27 (2013)

[11] Abdullah, Z., Herawan, T., Deris, M.M.: Detecting Critical Least Association Rules in Medical Databases. In: International Journal of Modern Physics: Conference Series, vol. 9, pp. 464–479. World Scientific Publishing Company (2012)

[12] Szathmary, L., Valtchev, P., Napoli, A.: Generating Rare Association Rules Using the Minimal Rare Itemsets Family. Int. J. Software Informatics 4(3), 219–238 (2010)

[13] Wang, K., Hee, Y., Han, J.: Pushing Support Constraints into Association Rules Mining. IEEE Transactions on Knowledge and Data Engineering 15(3), 642–658 (2003)

[14] Abdullah, Z., Herawan, T., Deris, M.M.: Tracing Significant Information using Critical Least Association Rules Model. International Journal of Innovative Computing and Applications, Inderscience 5, 3–17 (2013)

[15] Abdullah, Z., Herawan, T., Deris, M.M.: Mining Significant Least Association Rules Using Fast SLP-Growth Algorithm. In: Kim, T.-h., Adeli, H. (eds.) AST/UCMA/ISA/ACN 2010. LNCS, vol. 6059, pp. 324–336. Springer, Heidelberg (2010)

[16] Hoque, N., Nath, B., Bhattacharyya, D.K.: An Efficient Approach on Rare Association Rule Mining. In: Bansal, J.C., et al. (eds.) Proceedings of Seventh International Conference on Bio-Inspired Computing: Theories and Applications (BIC TA 2012). AISC, vol. 201, pp. 193–203. Springer, Heidelberg (2013)

[17] Tsang, S., Koh, Y.S., Dobbie, G.: Finding Interesting Rare Association Rules Using Rare Pattern Tree. In: Hameurlain, A., Küng, J., Wagner, R., Cuzzocrea, A., Dayal, U. (eds.) TLDKS VIII. LNCS, vol. 7790, pp. 157–173. Springer, Heidelberg (2013)

[18] Ding, J.: Efficient Association Rule Mining among Infrequent Items. Ph.D. Thesis, n University of Illinois at Chicago (2005)

[19] Liu, B., Hsu, W., Ma, Y.: Mining Association Rules with Multiple Minimum Supports. In: Proceeding of ACM SIGKDD 1999, pp. 337–341 (1999)

[20] Brin, S., Motwani, R., Silverstein, C.: Beyond Market Basket: Generalizing ARs to Correlations. In: Proceedings of the 1997 ACM-SIGMOD International Conference on the Management of Data (SIGMOD 1997), pp. 265–276 (1997)

[21] Omniecinski, E.: Alternative Interest Measures for Mining Associations. IEEE Transaction on Knowledge and Data Engineering 15, 57–69 (2003)

[22] Lee, Y.-K., Kim, W.-Y., Cai, Y.D., Han, J.: CoMine: Efficient Mining of Correlated Patterns. In: The Proceeding of 2003 International Conference on Data Mining (ICDM 2003), pp. 581–584 (2003)

[23] Herawan, T., Abdullah, Z., Mohd, W.M.W., Noraziah, A.: CLAR-Viz: Critical Least Association Rules Visualization. In: The 5th International Conference on Advanced Science and Technology (AST 2013), Hiddenbay Hotel, Yeosoo, South Korea, April 26-27 (2013)

Music Emotion Classification (MEC): Exploiting Vocal and Instrumental Sound Features

Mudiana Mokhsin Misron[1,*], Nurlaila Rosli[1],
Norehan Abdul Manaf[1], and Hamizan Abdul Halim[2]

[1] Faculty of Computer and Mathemathical Sciences
Universiti Teknologi MARA
Malaysia
[2] Faculty of Technology Management
Open Universiti Malaysia
Malaysia
{mudiana,norehan}@tmsk.uitm.edu.my,
laila8805@gmail.com,
hamizan.abdhalim@tm.com.my

Abstract. Music conveys and evokes feeling. Many studies that correlate music with emotion have been done as people nowadays often prefer to listen to a certain song that suits their moods or emotion .This project present works on classifying emotion in music by exploiting vocal and instrumental part of a song. The final system is able to use musical features extracted from vocal part and instrumental part of a song, such as spectral centroid, spectral rolloff and zero-cross as to classify whether selected Malay popular music contain "sad" or "happy" emotion. Fuzzy *k*-NN (FKNN) and artificial neural network (ANN) are used in this system as a machine classifier. The percentages of emotion classified in Malay popular songs are expected to be higher when both features are applied.

1 Introduction

Music has become more and more important in human lives and the need to improve the development of music acquisition and storage technology keep on rising. Music is a super-stimulus for the perception of musicality, where musicality is a perceived aspect of speech that provides information about the speaker's internal mental state [1]. It is believed that violation of or conformity to expectancy when listening to music is one of the main sources of musical emotion [2]. Thus, it is essential to conduct research as to analyze the similarities among music pieces based on which music can be organized in groups and recommended to user with suitable tastes. According to [3], music classification studies have so far been done with the main focused on classifying music according to genre and artist style. Recently, the

[*] Corresponding author.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_51, © Springer International Publishing Switzerland 2014

affective or to be specific the emotion aspect of music has become one of the important criterions in music classification.

Music emotion classification (MEC) is part of music data mining and artificial intelligence (AI) area of science. According to oxforddictionaries.com, music can be defined as vocal or instrumental sound and its common elements are pitch, rhythm, dynamics and timbre. Whereas, emotion is refer to as a strong feeling deriving from one's circumstances, mood or relationship with others. Primary emotion classes are happiness, sadness, anger, surprise, disgust and fear [4]. Emotion in certain music can be classified by employing two main processes namely, signal modelling and pattern matching. Based on work done in [5], signal modelling is referred to method of translating music audio signal into a set of musical features parameters. While, pattern matching is the process of parameter sets discovery from memory which strongly matches the parameter set obtained from the input music audio signal. All of this process automatically carried out using AI machine classifier such as supervised vector machine (SVM), artificial neural network (ANN), decision tree and etc. [6].

Until recently, most of MEC is done by looking at features such as audio, lyrics, social tags or combination of two or more features as stated above [7-9]. However, there were only few studies on MEC that exploits features from vocal part of the song [8]. It has been proved that, the timbre of the singing voice, such as aggressive, breathy, gravelly, high-pitched, or rapping is often directly related to our emotion perception and important for valence perception [10] thus it is suggested that vocal timbre should be incorporated to MEC. This research is proposed, to develop emotion classification system for Malay popular music from the year of 2000-2013. The final system should be able to use musical features extracted from vocal part and background music of a song as well as able to classify the type of emotion in music. The system will be employing two classification techniques namely, artificial neural network and fuzzy k-NN in order to classify category of emotion in selected Malay popular music. The overall system has implying data mining classification algorithm and techniques based on "Soft Computing and Data Mining" technology.

The discussion in this paper is divided into five sections, where the first part explains the overall idea of this research. Part two illustrates the literature review where, the previous and related works is clarified. Part three describes the data collection. Part four explains about music emotion classification system setup, fuzzy k-NN, ANN training and testing and classification results of the study. Part five discuses conclusions and proposed future works.

2 Literature Review

Generally, there are four main important things that need to be considered and understood in audio based music emotion classifications. Numbers of factors might obstruct the construction of a database and the issues such as which emotional model or how many emotion categories should be used in order to generate data collection must first be decided before one can proceed to the initial phase of MEC [8]. Fig. 1 below illustrates the typical audio based MEC as taken from [11].

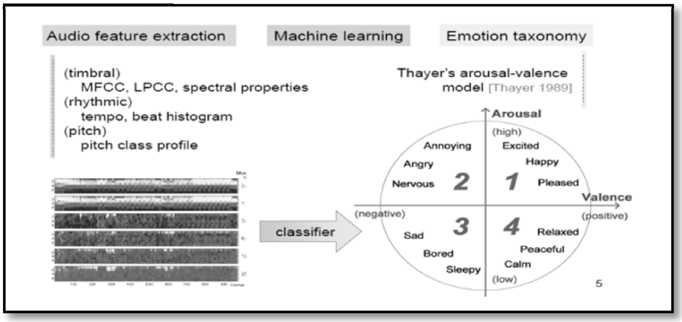


Fig. 1. Typical Audio Based Music Emotion Classification Taken from [11]

The growth in MEC have triggered the development of technology and system that related to emotion analysis and music signals, for example Moodtrack [12], Mood Cloud [13], and i.MTV [14]. These multimedia systems have applied MEC techniques which includes subjective test, musical features extraction, machine learning algorithm and etc. Normally, a subjective test is conducted to collect the ground truth data needed for training the computational model of emotion prediction. Subjective test can be done by numbers of annotation process, where selected annotators, manually listen to certain song and classify it based on group or classes.

Music listening is very subjective and multidimensional, especially in terms of emotions triggers. According to [15] and [16], different emotion insights of music are usually related with different patterns of acoustic cues. For example, excitement or feeling happy (arousal) is associated to tempo (fast/slow), pitch (high/low), loudness level (high/low), and timbre (bright/soft), where as sadness or valence is related to mode (major/minor) and harmony (consonant/dissonant) [17]. Generally, features of music such as timbre, rhythm, and harmony are extracted to signify the audio parameters of a music clips. Timbre is the characteristic of music stricture that can makes someone cry when a sad song being played to them. It is what makes a violin sound so beautifully sad and saxophone so blissfully happy. The timbre controls any emotion associated with the sound [18]. Several machine learning algorithms also applied to learn the relationship between music features and emotion labels.

The most used machine learning algorithms in MEC are artificial neural network (ANN). ANN has become one of the most significant mining techniques in various areas of science [19](Giudici,2003). The main reason for exploiting ANNs in those areas is because ANNs are very compatible as it can cater problem in various field and ANN is easy to manoeuvre as its operated just as same as human brain [20]. Another successful classification techniques in MEC besides ANN's, is by using the fuzzy k-NN (FKNN) classifier, as fuzzy logic is proved to be able to deal with uncertainty and imprecision in MEC [21]. Generally, two techniques has been used in this study, where the ANN classifier has been developed based on the overall MEC system, from training to testing, while, FKNN classifier technique has been developed based on work done in [22].

3 Data Collection

Due to the lack of ground truth data, most researchers compile their own databases [23]. Manual annotation is one of the most common ways to do this. For the purpose of this project, input music data are chose based on the result from subjective test that were carried out by categorizing Malay song into two main emotions, which is happy and sad. The rational of choosing only two emotions is because, happy and sad emotion is the basic emotion from psychological theories [24] and these two emotions cover a wide opposition of differentiation in 2D Rusell affect model representation with valence and arousal dimension [25]. "Happy" are positive valence and respectively high arousal, whereas "Sad" are in negative valance and respectively low arousal. So basically, both happy and sad quotient represents opposite value.

3.1 Subjective Test

Subjective Annotation test must been done in order to get all final 100 song with happy and sad emotion categorization. Previous studies have highlighted the important and common practice when doing this subjective test.

- Reducing the length of the music pieces [26][27].
- Providing synonyms to reduce the ambiguity of the affective terms [26].
- Using exemplar songs to better articulate what each emotion class means [28].
- Allowing the user to skip a song when none of the candidate emotion classes is appropriate to describe the affective content of the song [28].

For the purpose of this study, no restrict to exclusive categories will be compromised in order to undergone this subjective test. For each emotion, the problem is considered as binary classification. For example, one song can be categorized as either "happy or not happy" same goes to "sad or not sad". The dataset collection is made of 300 popular Malay song that is taken from Malay song charts from year 2000-2013 (Sources: Malay Radio Charts and "Anugerah Juara Lagu"). From this test, merely 50 songs that represent only "happy" emotion and another 50 songs represent only "sad" emotion will be allowed to be in the data collection for training purpose. To ensure the accuracy of the categorization process, 10 randomly selected annotators among teenagers with age range from 16-19 years old, were enquired to identify whether or not the data collection only contain "happy" and "sad" song.

3.2 Features Extraction

Music features extraction is the most crucial part in this study. It involves audio features extraction which has taking place as to determine the accuracy of data generation in the database. Generally, this project only focuses on timbre features which comprises of Spectral Rolloff, Zero-Cross, and Spectral Centroid.

Matlab programming is used to extract all of those selected features from every part of audio data (vocal part and instrumental part).

❖ *Spectral Rolloff*

Representation of the spectral shape of a sound and they are strongly correlated. It's defined as the frequency where 85% of the energy in the spectrum is below that frequency. If K is the bin that fulfills;

$$\sum_{n=0}^{k} x(n) = 0.85 \sum_{n=0}^{N-1} x(n) \tag{1}$$

Then the Spectral Rolloff frequency is f(K), where x(n)represents the magnitude of bin number n, and f(n) represents the center frequency of that bin.

❖ *Spectral Centroid*

Measure used in digital signal processing to exemplify a spectrum. It indicates where the "center of mass" of the spectrum is. Perceptually, it has a strong correlation with the impression of "brightness" of a sound. It is calculated as the weighted mean of the frequencies present in the signal, determined using a Fourier transform, with their magnitudes as the weights. Equation (2) is a formula to find the amount of spectral centroid in certain song.

$$Spectral \quad Centroid = \frac{\sum_{k=1}^{N} kF[k]}{\sum_{k=1}^{N} F[k]} \tag{2}$$

❖ *Zero-Cross*

Zero-Cross is the number of times a sound signal crosses the x-axis, this accounts for noisiness in a signal and is calculated using the following equation (3), where sign is 1 for positive arguments and 0 for negative arguments. X[n] is the time domain signal for frame t.

$$Z_t = \frac{1}{2}\sum_{n=1}^{N} | \, sign(x[n]) - sign(s[n-]) \tag{3}$$

4 Music Emotion Classification System

In order to classify two types of emotion to be exact, sad and happy emotion in selected Malay popular music, music data first must be converted into a standard format specifically; 22,050 Hz sampling frequency, 16-bits precision, 30 second frames. Overall process of MEC system from developing and testing are using MATLAB R12 programming language.

4.1 Fuzzy *k*-NN (FKNN) Classifier

Fuzzy k-NN (FKNN) classifier has implied combination of fuzzy logic and k-NN classifier. FKNN is widely used in pattern recognition. In [22], fuzzy membership μ_{uc} for an input sample $\mathbf{x_u}$ to each class $\mathbf{c}$ as a linear combination of the fuzzy vectors of k-nearest training samples. where μ_{ic} is the fuzzy membership of a training sample $\mathbf{x_i}$ in class $\mathbf{c}$, $\mathbf{x_i}$ is one of the k-nearest samples, and $\mathbf{w_i}$ is the weight inversely proportional to the distance $\mathbf{d_{iu}}$ between $\mathbf{x_i}$ and $\mathbf{x_u}$:

$$\mu_{uc} = \frac{\sum_{i=1}^{k} w_i \mu_{ic}}{\sum_{i=1}^{k} w_i} \tag{4}$$

$$w_i = d_{iu}^{-2} \tag{5}$$

With Eq. (4), we get the C×1 fuzzy vector μ_u indicating music emotion strength (C = 2) of the input sample:

$$\mu_u = \{\mu_{u1}, \ldots, \mu_{uc}, \ldots, \mu_{uC}\}^t \tag{6}$$

$$\sum_{c=1}^{C} \mu_{uc} = 1 \tag{7}$$

According to [22], the fuzzy vector of the training sample μ_i is computed in fuzzy labeling section. Several methods have been developed in [21] and [29], where $\mathbf{v}$ is the voted class of $\mathbf{x_i}$, $\mathbf{n_c}$ is the number of samples that belong to class $\mathbf{c}$ in the K-nearest training samples of $\mathbf{x_i}$, and $\boldsymbol{\beta}$ is a bias parameter indicating how $\mathbf{v}$ takes part in the labeling process **(β∈ [0,1])**. Different **β** is used during cross validation process, (**β**=0.0, 0.25, 0.50, 0.75, 1.0). When **β=1**, this is the crisp labeling that assigns each training sample full membership in the voted class **v**. When **β=0**, the memberships are assigned according to the K-nearest neighbors. The equation can be generalized as:

$$\mu_{ic} = \begin{cases} \beta + (n_c / K) * (1-\beta), & \text{if } c = v. \\ (n_c / K) * (1-\beta), & \text{if } c \neq v. \end{cases} \tag{8}$$

4.2 Artificial Neural Network (ANN)

The concept of Artificial Neural Networks (ANN) is based on biological neural networks. Neural network approaches have shown to be promising in supporting fundamental theoretical and practical research in artificial intelligence [20].

4.2.1 Neural Network Training

The network architecture used in this research is the feed forward back propagation. Neural network toolbox in MATLAB was utilized for training the neural network. It includes several variations of the standard back propagation. A variable learning rate that is a combination of adaptive learning rate and momentum training is used to train music clips data. 100 vocal audio data (comprises with 50 "sad" and 50 "happy" songs) and another 100 instrumental audio data (also comprises with 50 "sad" and 50 "happy" songs) data were used to train the neural network. All training data were in the standardized audio format. Training data was obtained from various sources in the internet and Malaysia's radio station. All of this audio data are split into 30 second frames.

4.2.2 Neural Network Testing

Testing process in MEC take place after database comprises with musical features are generated. Music data says for example "*Ombak Rindu"* one of the Malay popular songs is entered to the system. Automatically system will extract musical features from that particular song before ANN classifier can classify category of emotion contained in the song.

During the classification process, ANN classifier will get the information from the database or (memorized value of musical features) from previous training process. ANN classifier then can classify emotion from the song by scheming the music features vector as to produced result that close to 1 (happy) or close to 0 (sad).

As shown in Fig. 2, songs with an output ranging from 0.5≤x≤0.9 were considered as happy songs, while songs with output less than 0.5≥x≥0 were considered as sad songs. These tests were further verified using neural networks.

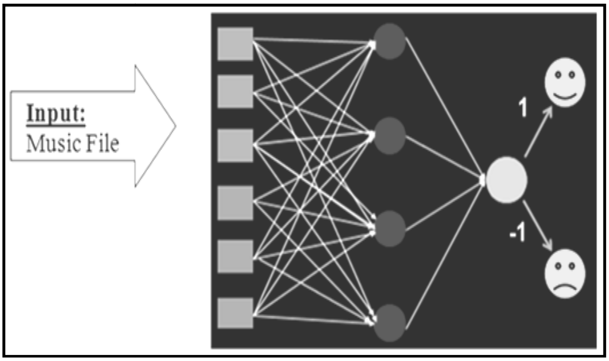


Fig. 2. ANN model for audio data testing

4.3 Testing Using Different Data

For system performance and classification accuracy evaluation, both classifier FKNN and ANN is tested with different data features. The testing process can be done using three different algorithms. This is to see the differences in classification rate when using different data. The details of algorithm used are as follow:-

- FKNN/ANN + Only Vocal Features
- FKNN/ANN + Only Instrumental Sound Features
- FKNN/ANN + Vocal + Instrumental Sound Features

4.4 Experimental Result

4.4.1 Cross Validation Result for Fuzzy k-NN Classifier Using Different β

Table 1. FKNN Classifier Using Different β

β	*Happy→Happy*	*Sad→Sad*	*Average*
0.0	78%	43%	60.5%
0.25	81%	46%	63.5%
0.50	72%	46%	59%
0.75	84%	57%	70.5%
1.0	87%	51%	69%

4.4.2 Result Classification Accuracy

30 songs that were categorized as happy song and the other 30 songs categorized as sad song were used to test the algorithm. Summary of the results is shown in Table 2 and Table 3.

Table 2. Test Results

Description	*No. of Data*	*Using FKNN %*	*No. of Data*	*Using ANN%*
Happy song	30	100	30	100
Classified as Happy Song	15	50	26	86.6
Sad Song	30	100	30	100
Classified as sad song	16	53.3	24	80

The accuracy of the classification result can be measured by dividing number of correctly classified songs with the total number of songs. A comparison of the accuracy of using only vocal features and the combination of vocal and instrumental sound features is shown in Table 3. The tests were administered using the same set of test music. Results show that the proposed approach which is using both data is more competitive than using only vocal or instrumental features as training data.

Table 3. Classification Rate Using Different Training Data

Algorithm	*Using FKNN %Accuracy*	*Using ANN %Accuracy*
Only Vocal Features	51	72
Only Instrumental Sound Features	52	75
Vocal+ Instrumental Sound Features	53.3	83.3

Based on the results, the accuracy of the algorithm is higher (more than 80%) when using ANN classifier for both vocal and instrumental sound features. Whereas, the accuracy of the algorithm using FKNN shows quite positive results although only able to classify slightly half of the selected song with exact emotions. It is shown that, the highest accuracy is at 70.5% when using **β**= 0.75.

5 Conclusion and Future Works

The music classification algorithm developed is proven to be up to 80% accurate using ANN techniques, while, the percentages of emotion successfully classified using FKNN is approximately 50%. The manoeuvring of vocal and instrumental features with the assistance of ANN classifier can provide successful music emotion classification. Data from timbre extraction for both vocal and instrumental sound is used as training data to the neural network. Vocal and instrumental sound features were combined to improve testing and classification accuracy. ANN learns to recognize emotion in music based on timbre musical texture as exist in the database. The system is developed through learning rather than programming. However, ANN is still unpredictable. It may take some time to learn a sudden drastic change. As for the fuzzy k-NN classifier, generally FKNN classifier has successfully classified songs in regards to certain group of emotions though the results not as high as when using ANN.

5.1 Future Works

Overall, this project has been manipulating two basic emotions as to categorize emotion in selected music (happy and sad affects). Besides, this work only focus on extracting timbre vectors in the music data, in which previous studies have recommend that timbre can be used to strongly determined the emotion or behaviour in both vocal and instrumental sound data. As for machine classifier, this project has used one of the most well-known artificial intelligence machines learning to be precise, Artificial Neural Network (ANN) and also fuzzy classifier (FKNN). Both of these techniques had been proved to be able to generate positive result, as expected. However, for future study, it is suggested that another types of music excerpt such as pitch, energy, harmony and etc; can be used to improved musical features database for training and testing process. With the positive result congregates from this project, it is extremely recommended if other types of emotion be considered as part of the classification category. This will hope to improve music emotion classification in the future.

Acknowledgments. This work is supported by a grant from the RMI, Mara University of Technology, Shah Alam.

References

1. Dorell, P.: What is Music?: Solving a Scientific Mystery, 318 p. NZ Publishing, Wellington (2005)
2. Imbrasaite, V.: Absolute Or Relative? A New Approach To Building Feature VecTors For Emotion Tracking In Music. In: Luck, G., Brabant, O. (eds.) Proceedings of the 3rd International Conference on Music & Emotion (ICME3), Jyväskylä, Finland, June 11-15 (2013)
3. Hu, Y., Chen, X., Yang, D.: Lyric-based song emotion detection with affective lexicon and fuzzy clustering method. In: Proceedings of the International Conference on Music Information Retrieval (2009)
4. Picard, R.W., Vyzas And, E., Healey, J.: Toward machine emotional intelligence: Analysis of affective physiological state. IEEE Trans. Pattern Anal. Mach. Intell. 10, 1175–1191 (2001)
5. Gilkes, M., Kachare, P., Kothalikar, R., Pius, V., Pednekar, R.M.: MFCC-based Vocal Emotion Recognition Using ANN. In: International Conference on Electronics Engineering and Informatics (ICEEI 2012) IPCSIT, vol. 49 (2012)
6. Lartillot, O., Toiviainen, P.: A Matlab Toolbox for Music Information Retrieval. In: Preisach, C., Burkhardt, H., Schmidt-Thieme, L., Decker, R. (eds.) Data Analysis, Machine Learning and Applications, Studies in Classification, Data Analysis, and Knowledge Organization, pp. 261–268 (2008)
7. Hu, Y., Chen, X., Yang, D.: Lyric-based song emotion detection with affective lexicon and fuzzy clustering method. In: Proceedings of the International Conference on Music Information Retrieval (2009)
8. Yang, Y.H., Chen, H.H.: Machine Recognition of Music Emotion: A Review. ACM Transactions on Intelligent Systems and Technology 3(3), Article 40 (2012)
9. Xu, M., Duan, L.-y., Cai, J., Chia, L.-T., Xu, C.S., Tian, Q.: HMM-based audio keyword generation. In: Aizawa, K., Nakamura, Y., Satoh, S. (eds.) PCM 2004. LNCS, vol. 3333, pp. 566–574. Springer, Heidelberg (2004)
10. Turnbull, D., Barrington, L., Torres, D.: Semantic annotation and retrieval of music and sound effects. IEEE Trans. Audio, Speech Lang. Process. 16(2), 467–476 (2008)
11. Yang, Y.-H., Lin, Y.-C., Cheng, H.-T., Liao, I.-B., Ho, Y.-C., Chen, H.H.: Toward multi-modal music emotion classification. In: Huang, Y.-M.R., Xu, C., Cheng, K.-S., Yang, J.-F.K., Swamy, M.N.S., Li, S., Ding, J.-W. (eds.) PCM 2008. LNCS, vol. 5353, pp. 70–79. Springer, Heidelberg (2008)
12. Vercoe, G.S.: Moodtrack: practical methods for assembling emotion-driven music. M.S. thesis, MIT,Cambridge, MA (2006)
13. Laurier, C., Herrera, P.: Mood cloud: A real-time musicmood visualization tool. In: Proceedings of the Computer Music Modeling and Retrieval (2008)

14. Zhang, S., Qingming, H., Qi, T., Shuqiang, J., Wen, G.: i. MTV: an integrated system for mtv affective analysis. In: Proceedings of the 16th ACM International Conference on Multimedia, pp. 985–986. ACM (2008)
15. Krumhansl, C.L.: Music: A link between cognition and emotion. Current Directions in Psychological Science 11(2), 45–50 (2002)
16. Juslin, P.N.: Cue utilization in communication of emotion in music performance: relating performance to perception. Journal of Experimental Psychology: Human Perception and Performance 26(6), 1797 (2000)
17. Gabrielsson, A., Erik, L.: The influence of musical structure on emotional expression (2001)
18. Lakatos, S.: A Common Perceptual Space for Harmonic and Percussive Timbres. Perception & Psychophysics 62(7), 1426–1439, PMID 11143454 (2000)
19. Giudici, P.: Applied Data Mining: Statistical Methods for Business and Industry. John Wiley & Sons, Inc. (2003)
20. Zurada, J.K.: Introduction to Artificial Neural Systems, 2nd edn. West Publishing Company (2006)
21. Keller, J.M., Gray, M.R., Givens, J.A.: A fuzzy k-nearest neighbor algorithm. IEEE Transactions on Systems, Man and Cybernetics (4), 580–585 (1985)
22. Yang, Y.H., Liu, C.C., Chen, H.H.: Music emotion classification: a fuzzy approach. In: Proceedings of the 14th Annual ACM International Conference on Multimedia. ACM (2006)
23. Yang, D., Lee, W.S.: Disambiguating Music Emotion Using Software Agents. In: ISMIR, vol. 4, pp. 218–223 (2004)
24. Juslin, P.N., Sloboda, J.A.: Music and emotion: Theory and research. Oxford University Press (2001)
25. Russell, J.A.: A circumplex model of affect. J. Personal. Social Psychol. 39(6), 1161–1178 (1980)
26. Skowronek, J., Mckinney, M.F., Van De Par, S.: A demonstrator for automatic music mood estimation. In: Proceedings of the International Conference on Music Information Retrieval (2007)
27. Yang, Y.-H., Lin, Y.-C., Cheng, H.-T., Liao, I.-B., Ho, Y.-C., Chen, H.H.: Toward multi-modal music emotion classification. In: Huang, Y.-M.R., Xu, C., Cheng, K.-S., Yang, J.-F.K., Swamy, M.N.S., Li, S., Ding, J.-W. (eds.) PCM 2008. LNCS, vol. 5353, pp. 70–79. Springer, Heidelberg (2008)
28. Hu, Y., Chen, X., Yang, D.: Lyric-based Song Emotion Detection with Affective Lexicon and Fuzzy Clustering Method. In: ISMIR, pp. 123–128 (2008)
29. Han, J.H., Kim, Y.K.: A Fuzzy K-NN Algorithm Using Weights from the Variance of Membership Values. In: CVPR (1999)

Resolving Uncertainty Information Using Case-Based Reasoning Approach in Weight-Loss Participatory Sensing Campaign

Andita Suci Pratiwi and Syarulnaziah Anawar

Universiti Teknikal Malaysia Melaka,
Faculty of Information and Communication Technology,
Durian Tunggal, Melaka, Malaysia
anditapratiwi@gmail.com, syarulnaziah@utem.edu.my

Abstract. Participatory sensing (PS) is an approach to distribute data collection, to analyze and interpret it. Identifying trusted and recommended participants with intention to have quality data to be analyzed is still a challenge because unlike in other domains, participatory sensing participants must fulfill the requirements of service provider, where participants are required to contribute quality data in a longer time frames. Many factors can influence the integrity of the information.One of major concerns in data contribution is the possibility of data truthfulness of being uncertain due to incompleteness, imprecision, vagueness, fragmentary. Consequently, it will cause the information to become unreliable to be analyzed. Detecting the uncertainty information is essentialto value the information. Therefore, the objective of this paper is two-fold. First, we give an overview of uncertainty information and the characteristics that suits participatory sensing system. Second, we outline how Case-based Reasoning approach can be implemented to tackling the uncertainty information in order to distinguish trusted and un-trusted participants.To address both objectives, this paper proposed uncertainty information detection approach based on information relevance using decision tree that integrate Case-based Reasoning, data mining, and information retrieval into our participatory sensing application, w8L0ss

Keywords: Uncertainty information, participatory sensing, case-based reasoning, weight-loss.

1 Introduction

Human plays significant role as the owner of the devices. The growth of smartphone era makes a new paradigm that smartphone is not only a device, but also has a capabilities of capturing moments as data through text, images, video and audio recordings, and GPS.Participatory sensing(PS) is an approach to distribute data collection, analyze and interpret it. Utilizing this new phenomenon, participatory sensing is a concept of human being a sensor, whether they act alone or in a group by sharing data that is captured by their smartphones to improve quality of life.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_52, © Springer International Publishing Switzerland 2014

Participants make their own decision about things that their sense ranging from health to culture [13]. It is much easier to collect amount of data in various scale using participatory sensing.Many application has been built to elaborate this paradigm, to collect more data to be analyzed. In participatory sensing, the participant is aware of the application's function because some service provider requires personal information of participant, for the purpose of demographic study or to offer financial interest such as incentive. This approach incorporates people into substantial decision of the sensing system, such as what data is shared and what extent of privacy mechanisms should be allowed to impact data correctness that had been decided by service provider. Later, data will be considered as information because data contain information.

While most research focus on assuring the trustworthiness of the system or network to prevent corruption or missing information during sharing time, we are exploring a different direction. This paper attemps to assure the trusworthiness of the received information after sharing time, where information might be uncertain if the information is incomplete due to human error or human nature behavior as the information's sources.

The rest of this paper is organized as follows: First, we give an overview of uncertainty information and its characteristics that is tailored to participatory sensing system. Next, we specify the theoretical basis for intelligent approach in resolving information uncertainty issues by presenting related work in the areas and outlining techniques in case-based reasoning (CBR) approach which will be used in our study. We then presented the implementation details of our weight-loss participatory sensing application, w8l0ss, and outline how CBR approach can be implemented to tackling the uncertainty information in order to distinguish trusted, recommended and untrusted participants. Finally, we conclude our findings and brief ourfuture work.

2 Uncertainty Information in Weight-Loss Participatory Sensing

Participatory sensing campaigns seek individuals willing to collect data about a particular phenomenon. A recruitment service takes campaign specificationsas input and recommends participants for involvement in data collections. Campaign specifications may involve a number of factors including participants' devicecapabilities, demographic diversity, and social network affiliation [11]. However, since participatory sensing is organized virtually, identifying trusted and recommended participants with intention to have good data to be analyzed is still a challenge.

Determining the quality of data being input by participants is important in participatory sensing participant's recruitment. Recruitement process needs to be based on the participants action and determine whether the participants is trusted or not. By learning from the old participants activities, we can analyze which participants could be recommended in the next participatory sensing campaign.

Unlike other domains, in participatory sensing, participants must fulfill the requirements of service provider in longer time frame. Service provider usually demands participants to continously input their information to the system. Many factors can influence participants while sharing their information, such as indolent feelings orparticipants getting bored to contribute because of the long period. This will instigate the participant to input random information or prevaricate to system.

2.1 Uncertainty Information

Information is the main focus in every system. If the information is of poor quality to be analyzed, then it affects to trushworthiness of information. Integrity of information should be maintained to hold the quality of information. Information has possibilities of being uncertain due to many grounds. Randomness, imprecision, incompleteness, vagueness, partial ignorance has been described in [5] as possible causes of information uncertainty. Uncertainty and incompleteness exist in every application domain [16]. Those possibilities of uncertainty information leads to untrusted information.

Uncertainty information is a complex and challenging issues and affects decision making in participatory sensing area. It impacts to trushworthiness of information while the information will be used for quality life improvement because of its uncertainty information might lead service provider to a wrong decision making. Uncertainty are divided into two (2) : *aleatory uncertainty* and *epistemic uncertainty*. *Aleatory uncertainty* is uncertainty of nature; the uncertainty being affected by nature and physical world. On the other hand, *epistemic uncertainty* comes from human's lacks [9].

To demonstrate how uncertainty can affects quality of information in participatory sensing system, we gather the characteristics of uncertainty information form the literature and determine which characteristics are suitable to participatory sensing area. In 2002, [3] explained a lot of types uncertainty : Fuzziness, ambiguity, inconsistency, permanent exceptions, temporary exceptions, limited validity, multiple options, nondeterminism, obscurity, vagueness, faults, rounding, null value, noisy data, etc. [7] add four other characteristics of become uncertainty, when it is incomplete, imprecise, fragmentary, unreliable, vague or contradictory.

2.2 Uncertainty Information in Weight-Loss Participatory Sensing Applications

According to World Health Organization (WHO) report in 2011, Malaysia has the highest rate of obese peopleamong Southeast Asian Nation [15]. Malaysian teenagers forecasted being obese at young age in 2020. Obesity can lead to other coroner diseases such as heart diseases, diabetes, etc. Realizing the importance to combat obesity problem, this paper will focus on participatory sensing application for weight-loss participatory campaign.

Our aim is to help people, especially obese people to reduce their weight and have better quality of life. The system helps them to control their food and their ideal calories in-take each day in terms of doing healthy diet and reduce weight periodically. To achieve the goal of the system, the requirements should be fulfilled by participants. Participants must input their daily routine (What they eat? Do they workout?) continously until the campaign ends. While participants contributing, the system collects amount of data to be analyzed to give feedback for further system performance improvement. In the mean time, due to data correctness issue, quality of data is an important thing to be preconcerted especially the data integrity before analyzing the information itself.

It is not easy to detect uncertainty of information, specifically in large amount at the same time and continuously. Detecting the uncertainty information is necessary to value the information to determine the participants' behavioral action. At the end of the campaign, it will give the service provider dataset of trusted and recommended participants.

This paper study incompleteness, relevance and rounding as characteristics that influence to uncertainty information in participatory sensing. To further understanding each characteristic, the present study define explication of each characteristic that suitable to weight-loss participatory sensing application with instance on it. (See following Table 1):

Table 1. Characteristic Instances For Weight-Loss Participatory Sensing Application

Characteristic	Instance
Incompleteness	Participant should input in the end of week but some participants only input twice in a month.
Relevances	BMI data is related to height and weight.
Rounding	“Someone inputting their calorie intake is 10000”. It may be less or more than that.

Investigating characteristics that influence to uncertainty information is essential in order to know which characteristics are suitable to participatory sensing system because not all of characteristics can be used to detect uncertainty information from participants contributed data. From the study, three characteristics of uncertainty is found:*incompleteness* and *relevance* which previously has been studied, and *rounding* which has never been used before.

3 Intelligent Approach for Uncertainty Information Detection

3.1 Related Work

Liang et al [10] studied uncertainty to develope a novel methodology of fuzzy inferenced decision making (FIND) that solved the problem of decision making under uncertainty and incompleteness. They combined dual mode fuzzy belief state base and dual state fuzzy association as a new reasoning paradigm. FIND was tested in medical diagnosis application.

Relevance is a characteristic studied by Lalmas [8]. In the study, relevance is explained as a statement “of the less relevance is, the more uncertainty the information is". Lalmas studied relevance as characteristic in his work to constructed information retrieval model that aim to captured uncertainty as essensial feature of information but the result showed that the performance was not satisfactory. Relevance is also studied by Nottelmann et al [11] using probability technique. They studied probability of relevances for advanced information retrieval application from uncertainty inference. of probability : linear function and logistic function. The result showed that the probability of relevances can be achieved but it was slightly improved by using logistic function. The same approach is taken by Wolf et. Al [16], that solve relevance problem to handle uncertainty under incomplete database.

Decision tree is a common technique in data mining for classification, it is used to supports decision making. Decision tree is popular because its easy to understand with readable rules [14]. Constructed decision tree classifier on uncertain numerical attributes data using Distribution-based approach to compare with Averaging approach, in order to know which approachcould lead to a higher classification accuracy. They also established a theoritical foundation on pruning technique to significantly improved the computational efficiency of the Distribution-based algorithm. The result showed that the classical decision tree have been modified and exploiting data uncertainty leads to decision trees with higher accuracies but the performance is still an issue. They also devised a set of pruning technique and improve the tree construction efficiency, which was experimentally verified to be highly effective.

3.2 Proposed Solution: Uncertainty Information Detection Using Case-Based Reasoning

In most research, CBR is used for decision making process or medical diagnosys but its rationally can be used for recognizing human behaviour by learning other people behavior who have similarity in action. In participatory sensing recruitment process, it is important to know which participant can be trusted to be the information source and will lead to campaign's goal accomplishment. By learning from dataset of older participants or even other participants in the same campaign, CBR approach can resolve the uncertainties of informationas a basis in determining the participant's reputation.

The present study integrates CBR, data mining, and information retrieval approach for detecting uncertainty information in our weight-loss participatory sensing application; w8l0ss. The application collects participant's input and store it in databases. Knowledge miner is built based on databases and guidelines to retrieve participant cases. A new participant with a new cases will trigger the application to find the similar cases and make case adaptation to decide the best case solution for the new case. The solved case is saved as a new case in participant cases and decide whether the participant is trusted and recommended or participant should be drop out.

In this study, CBR mechanism will be applied to find the similar case by retrieving the previous cases and make an adaptation for the new case. From the adapted case, the solution from the cases will be used to the new case or if it needs then the solution can be revise to have the best solution for the new case. Once the new case solve, the case and solution will be retain in database to be used in the future. The proposed system architecture is shown in the Figure 1.

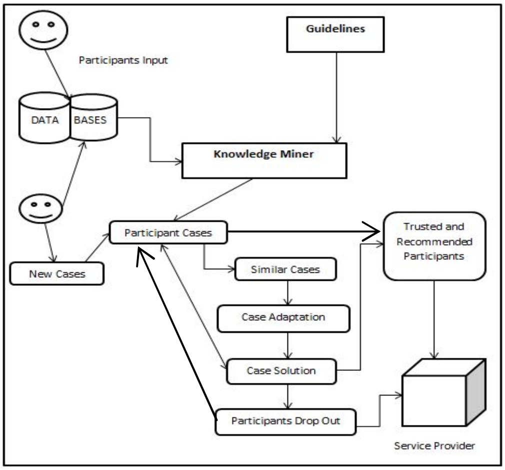


Fig. 1. Architecture of Uncertainty Information Detection using Case-Base Reasoning in Project w8l0ss

3.2.1 CBR Cycle

Case-Based Reasoning is an approach to solve a new problem by learning from old problems and try to suggests the best solution from old cases. The cycle of Case-Based Reasoning consists of four procedures : Retrieve, Reuse, Revise and Retain. First, case retrievalis wherethe previous cases are first retrieved andlater are judged to find similarities with the new case. Second phases is case adaptation or reuse the similar case. In this step, the old solution case is reused as an inspiration for solving new cases.

A new case may not exactly match the old one, but the old knowledge often need to be fixed to fit the new one. The third step is revise, this is the step when the old knowledge has fixed to the new one and applied to solving the new problem [6]. Then, the selected case is reused by copying or integrating the solutions from the cases retrieve. Next, solution is integrated and revised or by adapting the solution retrieved in an attempt to solve the new case. the last phase is retaining the new solution if it has been validated [10].

4 Implementation Details

4.1 System Architecture

We develop Project w8L0ss [1], [2], a participatory sensing application which is initiated for organization i.e., service provider, to run a wellness-program. It allows participants to record their activities that contributed to weight-loss. The activities include self-weight monitoring, dietary recording, physical activity recording, and coaching. w8L0ss is developed under Android platform. To support data collection, w8L0ss embedded autonomous engagement at various modes. The application is currently undergoing testing.

4.2 Experimentation Variables

This work suggests that uncertainty information detection operates by enhancing focus on the relevance of the information which is determined based on the following metrics [2]:

4.2.1 *Participant Input (f)* = The quantity of participation is measured by the frequency of input recorded by the participants. Only participants who input 2 or more in at least one month will accounted for data collection.

$$Frequency = f > 2$$

4.2.2 *Targeted weight (O)* = To accomplish the goal of the campaign, participant must record their target weight as a requirement to be analyzed whether the number is achievable or not.

4.2.3 *Weight (W)* = the variable calculated by the initial and final weight recorded. In the experiment the following formula will be used :

$$Weightloss = \frac{InitialWeight - FinalWeight}{FinalWeight}\ x\ 100 \tag{1}$$

4.2.4 *Goal accomplishment (tg)* = To have the quality of participation, the appropriates of goals was evaluated and the number of it served as the indicator for quality participation anad it considered appropriate if the targeted goal using guidelines suggested by [15]. For goals variables, the data is coded using the following formula:

$$\boldsymbol{Goals} = \left\{\frac{\boldsymbol{Weightloss}}{\boldsymbol{Initial\ Weight - Targeted\ Weight}} \quad \begin{matrix}\mathbf{1}\ \boldsymbol{Goals} > 1 \\ \mathbf{0}\ \boldsymbol{Otherwise}\end{matrix}\right. \tag{2}$$

4.2.5 *Calorie(e)* : Calorie is an important thing needed by human body to live that will be made to be energy in where calorie obtained from food but overage calorie in human body can cause diseases. Physical activity expends calorie as energy out.

4.3 Relevances Identification for Uncertainty Information

In this section, we describe formulation of relevance identification to detect uncertainty information using decision tree and bayes techniques. The relevance of information is identified between each participants' input based on calories formulation. We exclude other characteristics identification due to unavailability of dataset. From [4], ideal calorie formula is:

$$Cmx = 0.99\ x\ W_1\ x\ 24\ x\ 1.55 \tag{3}$$

From (3), we derived relevances under two conditions: relevance of weight-loss information recorded each week by participant and relevance of targeted weight information inputted at the beginning of month. Relevance of weight-loss are formulated based on daily intake that should be below maximum daily calorie :

$$Relevances_{Weightloss} = \left\{ Cmx \left[\frac{(W1-Wn)x\ 7700}{d}\right] \begin{matrix} 1 & \leq & Cmx \\ 0 & Otherwise & \end{matrix} \right. \tag{4}$$

Then equation (5) is formulated for targeted weight relevances based on target weight-loss that should be below maximum weight-loss :

$$Relevances_{Target} = \left\{ Cmx \left[\frac{Cmx-600}{500}\ x\ 0.5\right] x\ 4 \quad \begin{matrix} 1 & \leq & tg \\ 0 & Otherwise & \end{matrix} \right. \tag{5}$$

4.4 Implementation of CBR Using Decision Tree

In this paper we implement the decision tree technique to know the relevances between variables. We used data from [2], a previous participatory weight-loss self-monitoring campaign in recording and charting their weight-loss progress towards the targeted weight. 37 participants have joined the program for duration of three months. Participants is deem as trusted participant if the information they shared is fully relevance (based on weight-loss result, weight-loss relevancy, targeted weight relevancy and frequency of inputs). Participant is deem partially trusted if only one or two of their information is relevant. Lastly, participant is untrusted if there is no relevancies in any of the information that they had shared.

From the implementation in the real data, the result shows from input of their weight-loss result, 36 participants are trusted eventhough 4 of them are gain weight, 4 others are stay on the same weight but 28 of participants are loss their weight and only one who determined as partially trusted participants. 33 participants are trusted based on weight-loss relevancy and calorie as parameter and 4 participant are partially trusted. Based on frequency of their input, the result shows 1 participant is partially trusted and 36 participants are trusted.

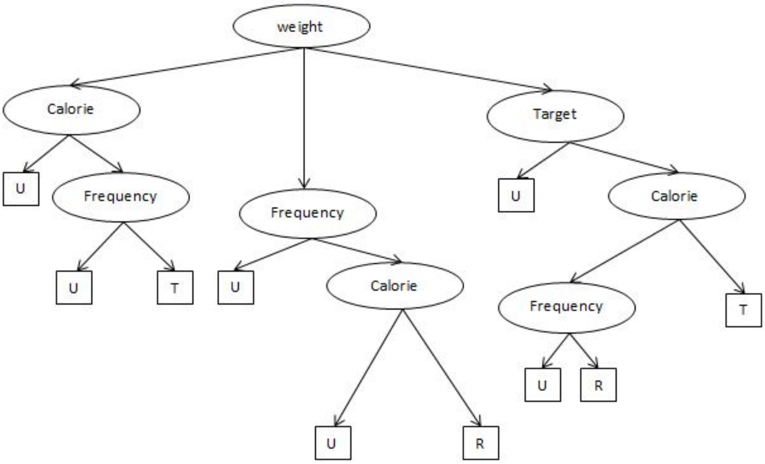
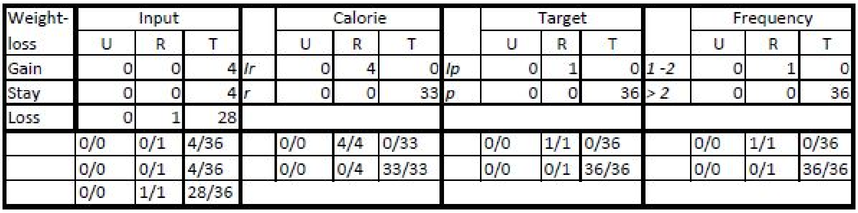


Fig. 2. The decision tree for relevances of each variables

Table 2. Probability of the case data



Weight-loss	Input				Calorie				Target				Frequency		
	U	R	T		U	R	T		U	R	T		U	R	T
Gain	0	0	4	*lr*	0	4	0	*lp*	0	1	0	*1 -2*	0	1	0
Stay	0	0	4	*r*	0	0	33	*p*	0	0	36	*> 2*	0	0	36
Loss	0	1	28												
	0/0	0/1	4/36		0/0	4/4	0/33		0/0	1/1	0/36		0/0	1/1	0/36
	0/0	0/1	4/36		0/0	0/4	33/33		0/0	0/1	36/36		0/0	0/1	36/36
	0/0	1/1	28/36												

5 Conclusion and Future Work

From the implementation, the result is obtained by using Decision Tree and Bayes Probability on real –data of weightloss campaign. We found both of Decision Tree and Bayes Probability are able to determined reputation of participants based on weight-loss result, weight-loss relevancy and targeted weight relevancy, and frequency of their input that should be at least 2 times.

Future study will use these techniques to solve others characteristics, Incompleteness and Rounding and applied CBR techniques to obtain the best result. Next, we will implement the CBR module to the real data of weightloss programme campaign.

Acknowledgments. This research is supported by Fundamental Research Grant Scheme, Ministry of Higher Education, Malaysia (FRGS/2012/FTMK/TK06/03/1-F00142).

References

1. Anawar, S., Yahya, S.: Empowering Health Behaviour Intervention Through Computational Approach for Intrinsic Incentives in Participatory Sensing Application. In: Proceeding of International Conference of Research Innovation in Information System, ICRIIS 2013, Bangi, Malaysia (2013)
2. Anawar, S., Yahya, S., Ananta, G.P., Abidin, Z.Z., Ayop, Z.: Conceptualizing Autonomous Engagement in Participatory Sensing Design: A Deployment for Weightloss Self Monitoring Campaign. In: Proceeding of IEEE Conference on e-Learning, e-Management, and e-Service, IC3E 2013, Kuching, Malaysia (2013)
3. Berztiss, A.T.: Uncertainty Management (2002)
4. Blackburn, G.L., Bistrian, B.R., Maini, B.S., Schlamm, H.T., Smith, M.F.: Nutritional and Metabolistic Assessment of the Hospitalized Patient. JPEN J. Parenter. Enteral Nutr. 1, 11–21 (1977)
5. Coppi, R.: Management of uncertainty in Statistical Reasoning: The case of Regression Analysis. International Journal of Approximate Reasoning 47(3), 284–305 (2008)
6. Huang, M., Huang, H., Chen, M.: Constructing a personalized e-learning system based on genetic algorithm and case-based reasoning approach. Expert Systems with Applications 33(3), 551–564 (2006)
7. Klir, G.J.: Uncertainty and Information: Foundation of Generalized Information Theory. John Wiley & Sons, Inc., Canada (2006)
8. Lalmas, M.: Information Retrieval and Dempster-Shafer's Theory of Evidence. In: Jul, E. (ed.) ECOOP 1998. LNCS, vol. 1445, pp. 157–176. Springer, Heidelberg (1998)
9. Li, Y., Chen, J., Feng, L.: Dealing with Uncertainty: A Survey of Theories and Practices. IEEE Transactions on Knowledge and Data Engineering 25(11), 2463–2482 (2013)
10. Liang, L.R., Looney, C.G., Mandal, V.: Fuzzy-inferenced decision making under uncertainty and incompleteness. Applied Soft Computing 11(4), 3534–3545 (2011)
11. Nottelmann, H., Fuhr, N.: From Uncertain Inference to Probability of Relevance for Advanced IR Applications. In: Sebastiani, F. (ed.) ECIR 2003. LNCS, vol. 2633, pp. 235–250. Springer, Heidelberg (2003)
12. Pal, S.K., Shiu, S.C.K.: Case-Based Reasoning. John Wiley & Sons, Inc., Canada (2004)
13. Reddy, S., Estrin, D., Srivastava, M.: Recruitment Framework for Participatory Sensing Data Collections. In: Floréen, P., Krüger, A., Spasojevic, M. (eds.) Pervasive 2010. LNCS, vol. 6030, pp. 138–155. Springer, Heidelberg (2010)
14. Tsang, S., Kao, B., Yip, K.Y., Ho, W.-S., Lee, S.D.: Decision Trees for Uncertain Data. IEEE Transactions on Knowledge and Data Engineering 23(1), 64–78 (2011)
15. Mohamad, W.N., Musa, K.I., Md Khir, A.S.: Prevalence of overweight and obesity among adult Malaysians: an update. Asia Pac. J. Clin. Nutr. 20(1), 35–41 (2011)
16. Wolf, G., Kalavagattu, A., Khatri, H., Balakrishnan, R., Chokshi, B., Fan, J., Kambhampati, S.: Query processing over incomplete autonomous databases: query rewriting using learned data dependencies. The VLDB Journal 18(5), 1167–1190 (2009)

Towards a Model-Based Framework for Integrating Usability Evaluation Techniques in Agile Software Model

Saad Masood Butt[1], Azura Onn[2], Moaz Masood Butt[3], and Nadra Tabassam[4]

[1] Computer and Information Sciecne Department,
Universiti Teknologi PETRONAS Tronoh, Perak, Malaysia
saadmasoodbutt668@yahoo.com
[2] Department of Management and Human Resource, Universiti Tenaga Nasional, Malaysia
azura@uniten.edu.my
[3] Computer and Software Engineering, Bahria University Islamabad, Pakistan
moazbutt786@hotmail.com
[4] Comsats Insititude of Information Technology, Wah Cantt, Pakistan
nadrainam@hotmail.com

Abstract. Various new agile software models were offered though agile manifesto as a counteraction to conventional and extensive software techniques and process design. SE followed a systematic approach of development. Whereas integrating usability in software development improved the ability of software product to be used, learned and be attractive to the users. Research showed the benefit of usability; yet, to this day agile software model continues to exhibit less importance of this quality attribute. Moreover, poor usability and inefficient design were the common reasons in software product failure. The aim of this paper was to develop a model to integrate usability evaluation methods into agile software model. This was done by proposing a unique model and evaluate the proposed model by using IEEE Std 12207-2008, ISO 9241:210.

Keywords: Software Development, Usability Evaluation, Agile Model, Software Models, Usability Engineering.

1 Introduction

The term usability is defined in various ways throughout the literature. Despite the difference in definition of usability, most of them classify usability as a quality attribute for the software success and make it usable to the users. Today, software are being developed for individual users as well as for companies. To capture the market and gain profit many software models were presented as agile manifesto just because of traditional rigorous software process models. Hence, this manifesto gives an idea of a new methodology in software development, which is well known today as agile software method. Agile software method does not follow documents as seem in traditional software methods. Instead, agile focus more on coding then documentation follows. The best thing in agile methodology is that, it resists rapid changes in

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_53, © Springer International Publishing Switzerland 2014

software development that is not easy in traditional software model. Still, agile model lacks a quality attribute of Usability in its development. Research [1][2][3] shows the high importance of usability in the software development. Poor and inefficient designs are the most common causes noted in the product that have lack of usability and ultimately result in the failure of soft product. So, neglecting usability from software development make the software less interactive, difficult to use and dissatisfy the user.

It is very obvious that successful software is always the result of good collaboration of different experts like software engineer, usability expert, software tester, stakeholders and users. But in reality, these experts do not cooperate as efficiently as significant in the software success [2]. Do to this lack of cooperation among experts; many software projects fail to deliver software on time with complete requirements. Common failures such as (a) late participation of user generates a high impact on the software; (b) CTR plan developed for software exceeded by a large factor; (c) software is not easy to use and understand as technical people are involved in the designing of software and (d) software delivery is not providing the good quality expected. There are many other factors of software project failure [2].

In this paper, a new agile software model is proposed with the integration of usability evaluation techniques. Later the proposed model is validated by using international standards i.e. IEEE Std 12207-2008, ISO 9241:210.

2 Literature Review

Many of the standard development methodologies are based on technical dimensions. But, they fall apart when trying to meet up with the project strategies. Therefore, the researchers have come up with a model that is standardized by ISO/IEC12207. It uses balanced score card to fulfil missing dimensions in the project strategies. Agile software development model is mapped with ISO/IEC12207 using balanced score card to create the appropriate action plan. A model [1] is developed that helps project managers to evaluate effectiveness of the Agile Software Development model using balanced score card. Balanced score card would ensure that the goals and objectives of the project are being met. Balanced score card aligns business strategies with the action plans. It helps project managers to align his/her decisions with the business strategies from the score card aligning people and activities to it.

ISO/IEC12207 is a standard for software life cycle processes for systems and software engineering to use its elements as an established set of Life Cycle Processes. It establishes conformance of the project to the established environment. It contains processes, tasks and activities that would be applied during the acquisition, supply, development, operations, maintenance and disposal of software. Agile Software Development method was developed by Agile Manifesto in 2001. It is based on customer's collaboration sharing requirements and values to be incorporated in the product. Highest priority is given to customer satisfaction. The scrum is adapted by agile software development methodology. It introduces iterations called sprints. Each sprint refers to a short term plan in which developers should meet on daily basis for discussions. Tasks are backlogged into scrum and each sprint is allocated a task to be done within a time frame. So, the agile software development is mapped on to the ISO/IEC12207:2008. It applies 27 processes to achieve the set targets. These processes are further mapped on to the balanced score card using Analytical Hierarchy Process (AHP). Hierarchy of decisions is developed from the measures of related processes. Firstly, software response time is measured, then decision criteria is formed that is mapped on to the goal achievement processes. And, other processes are prioritized accordingly.

McBreen [6] describes the concept of agile methods is to the rapid development of software focusing less time on analysis and design. Main focus is to get feedback from customer after every deliverable. This can be done by making development process fast and follow incremental and iterative approach to serve customer in efficient way. Though its incremental and iterative nature, do not support software interface designing which is mandatory for the development of usable and interactive software [7].

Kane [8] discussed that none of the agile methods are supporting the usability or explicitly incorporates usability in the agile methods. Also suggested to incorporate usability in agile methods will increase product usability and increase end user satisfaction.

Fox et al [9] addressed that no agile methods can ensure the usability of software or software is usable. For the development of interactive and useable software, agile needs to follow usability techniques in its development methodology.

Many software companies still using agile methods to develop software in small amount of time. But now they agree that agile alone can't ensure the usability of software. It needs to integrate with Usability [10]. But usability expert not sure that the resulting software is developed with the actual end-users participation or not [11]. Agile methods such as Extreme Programming (XP) value the customer who is present onsite during the development but still not clear where these are the actual user or representative of end users [12].

Sharp et al [13] present collaboration of customers in Extreme Programming (XP) and highlights the useful of UCD in XP. The major issues highlighted in his paper was the lack of trust between customers and developers. Participation of the right user not only help to get right requirement also increase the usability of software. Still customer cannot act like real users in the development of software [14]. Large scale software development companies employ waterfall model that is plan driven and takes long time to complete. This approach is susceptible to failures in rapidly changing environments. Thus, another approach is introduced that understands the dynamics of software development projects known as Agile Software Development Life Cycle. 'Agile genome' defines seven characteristics of agile projects. Systems Dynamics Model is constructed for agile software projects called Agile Project Dynamics (APD) model [2] that takes care of all seven aspects as a major component of the model. Many of the commercial software's used agile methodology to deal with the pressures that could occur in traditional development in terms of requirements changes, schedule delays, defects that result in endless delays and redesign.

Agile development methodologies are getting wide acceptance as they address many Software development risks. Faster delivery of software is made possible and it is flexible towards changes introduced in the software with time. Organizations adopting agile are inclined towards adding features that increase user interest in the system in terms of value and usability [3]. Usability engineering explores Human Computer Interaction (HCI) focusing on how people interact with the systems. But, it was difficult to integrate user interactive process in traditional agile methodologies as used in practice. This lead to the evolution of eXtreme Scenario-based Design (XSBD) [3] process that integrated agile usability approach. XSBD maps well on the established Scenario-Based Design (SBD) process already part of usability engineering fundamentals and is also in compliance with the agile development model using XP and Scrum. XSBD keeps large softwares on track by ensuring quality by system usability measure. Central Design Record (CDR) forms the core of XSBD, which provides sharing of design that guides usability process. Thus, usability and agile development work practices closely coordinate and communicate in XSBD. The usability evaluation results are coupled with the design and high level project goals adhering to the key benefits of SBD and links to the agile work process. XSBD has been developed and tested by partnering with several Software Development Companies and results gathered through practitioners who used XSBD in their development process. The results of this research demonstrate a broad scope of continued research in adopting it in practice and linking it with HCI methodologies and reusability of knowledge gained.

Here software development and usability design runs in parallel. Personals in usability and development team closely collaborate to come up with the quality system keeping to the quality standards with a focus on increased user interactivity. CDR tightly couples evaluation results generated through usability development process with the design features and goals of large systems making it possible for the usability engineer to embed key benefits of SBD while remaining within agile incremental development cycle. With partnering with companies like; Meridium, Inc. etc. helped refining XSBD approach with actual implementation of it in practice. In the analysis, the different divergent aspects were addressed between usability and agile methodology. Based on the case studies and analysis, the principles were formed for the practitioners who would follow agile usability approach.

Usability testing does not require long time span or a higher budget to be more effective. ‘Discount usability’ allows engineers work in team by thinking aloud, card sorting, scenario-based, walkthroughs and using heuristic approach – making the process much cheaper, fast and easy. These techniques can be applied early in the life cycle and during implementation phase for evaluating major/minor usability issues. In this framework, discount usability model is used within agile setting to be iterative and be more effective [5]. The software not only has to be useful but it also needs to be usable these days. Agile development model follow an iterative approach and has a very strict time frame where daily scrum meeting is held to update the team with the happenings. Researchers came up with possibility of merging usability methods with the agile model. Adopting an agile approach while focusing on usability centered design, lead to an awesome experience that resulted in timely delivery of a highly usable product. The motivation for using agile with usability measures was because agile environment had a tested procedure at any point in the development phase.

It was made for effective by inviting the User Experience Team to play as a customer to evaluate the usability of the product. The cycles shown in figure 5, could use any technique to gather user data like; scenario-based, or walkthroughs, etc. at the end of each cycle a working deliverable is expected featuring customer's expectations. The success of discount usability model is that it is [5]:

1. Easy to use, teach and comprehend. In just a half hour meeting the heuristics of usability techniques could be laid down to reveal issues that could be present in the product at hand.
2. The discount usability model is very cheap to adopt as no expensive tools or equipment is required.
3. No usability experts need to be hired to perform evaluation. Evaluating is a very flexible process in discount usability.
4. Using techniques like; card sorting can get early feedback in the design process before reaching to a working system.

Discount usability also as its limitations that should not be ignored [5]:

1. Over simplification of this method adds some distortion to it that is introduced in it while making it easy, and fast. When making discount usability out of traditional usability approach, only the key principles were adopted out of thousands of entries. Thus, it became more generalized and could confuse developers.
2. Although it is understood that evaluation process does not need to recruit usability experts or end-users making the procedure flexible, there is an opinion that this could lead to misinterpreted changes that are actually not required in real. And, that would degrade the usability of the system.

3 Survey Results

For this analysis, a survey has been conducted 45 randomly selected IT professionals from the Information Technology domain have participated in the survey. The purpose of the study is to create an "Agile Usability Software Engineering Life Cycle" [2] [3] [4] that could comprehend the influence of the Users in the software development process. The purpose is to make software development process reliable and finally integrate the Usability Evaluation to make the software more usable. The distribution of the survey targeted IT experts, researchers, software users and stakeholders. The questionnaire is divided into four sections. Section A is about demographic information of all those people who will answer the survey questions. Section B is on the software process particularly focusing agile process in software industry. Section C look into the Usability Evaluation in the agile software process. Section D is focused on developing a process that able to do the things that normally require human intelligence to perform that task in software development. All questions mentioned in every section and was rated using the scale of 1 to 5

(1= strongly disagree, 2= disagree, 3= fair, 4= agree, and 5= strongly agree). Figure 1 shows 81% respondents agreed on the active participation of users in software development. Also software interfaces play magnificent role in product success and failure but also prefer less documentation in software development process. Rest 5% disagree with the active participation of users in software development, software interfaces plays role in product quality and also disagree on less document in software development.

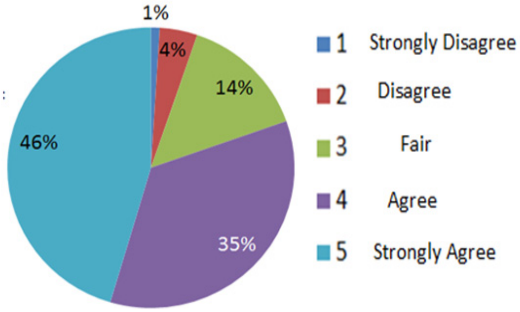


Fig. 1. Active Participation

From Figure 2 it shows 78% respondents agreed on the Evaluation methods should be considered in software development. Prefer to consider in evaluation methods in agile software development. Remaining 6% disagree with the points mentioned above.

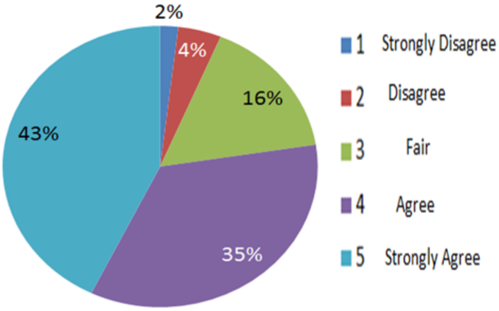


Fig. 2. Usability Evaluation in the agile software process

From Figure 3 it shows 82% respondents agreed on a development such process that is the part of the software model to make development faster. The other 4% disagrees with such model used in software process.

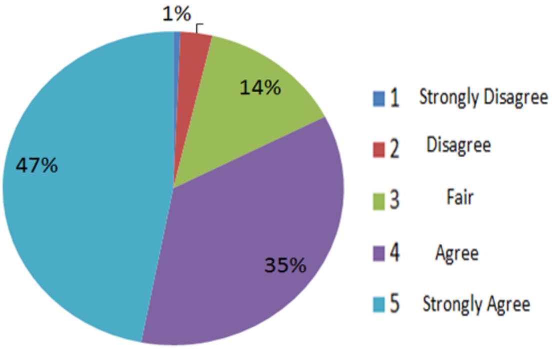


Fig. 3. Use various processes in an Agile Software Model for faster development

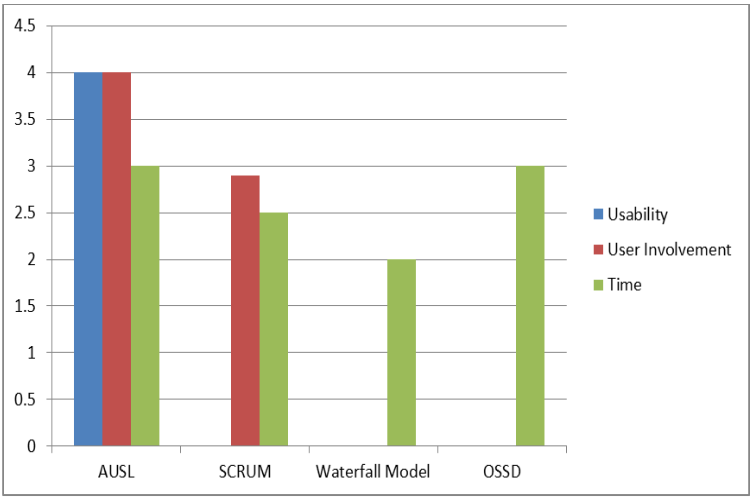


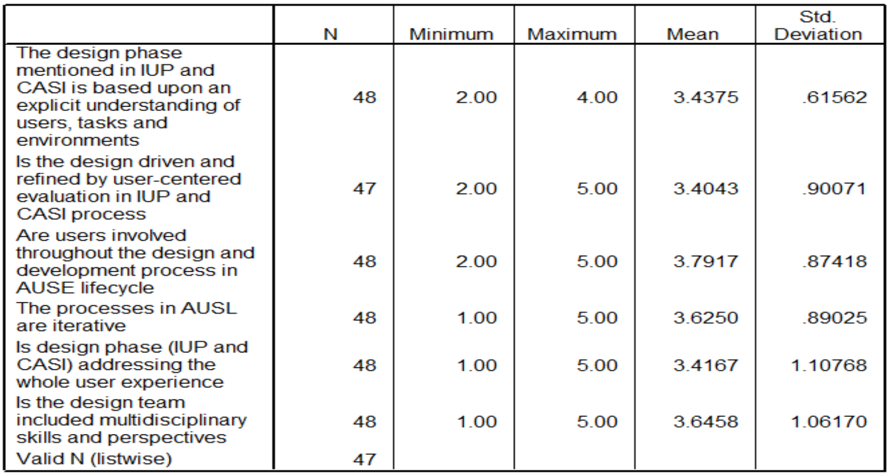
Fig. 4. Comparison with other Models

The focus of this paper mainly is to analyze the role of usability and users in the software model. From the research, it has been discovered that in the software development, the parts played by HCI and users are important. In addition role of helping process in software model that makes development faster and effective. After analysis of various software models through the numerous factors and keeping survey analysis report that discussed, it has been found that all models expect agile model are expensive to use (in term of cost, time and resources), used for big projects and having lack of Usability approaches. Whereas agile software model is a renowned model and is followed by many companies for small medium and large project. Hence introducing usability approaches in agile model increase the efficiency and usability of software. The same project subsequently was developed using the most popular methodologies such as Scrum, Waterfall Model and OSSD and was rated using the scale of 0 to 4 (0= No, 1= fair, 2= Good, 4= Excellent). The results shown in figure 4 are based on three important features; Usability, User Involvement in software development and Time taken to meet the deadline of the project deployment. From the outcomes it indicates that Usability concentrated more in AUSL as compared to other models. On the other hand, User Involvements was observed more in AUSL and Scrum. The time for completion outcome shows that by using OSSD the time of completion will be lesser as compared to other software models.

4 Validation

There are some essential features that mostly considered in validation of the software model. In the proposed AUSE lifecycle, Industry Standards should be followed to validate every process and make the processes of AUSE standardize. To validate AUSE life cycle, ISO 9241:210 [26] (Usability standards) and IEEE Std 12207-2008 (System Context Processes) will be followed. The International Standards (ISO 9241:210 and IEEE Std 12207-2008) determine a common model for software life cycle process, having a well-defined terminology that can be recommended by the software industry [20]. Table 1 shows the most common processes of system set by the (International Standards Group) that may be performed during the lifecycle of software system. The outcome mentioned in each process need to be achieved to standardize the process. 11 standard process are use in order to validate and standardize the AUSE life cycle then random survey was taken globally in which 49 respondent filled the survey form. Later SPSS tool was used to perform different analysis.

Table 1. Descriptive Analysis of Usability Standard



	N	Minimum	Maximum	Mean	Std. Deviation
The design phase mentioned in IUP and CASI is based upon an explicit understanding of users, tasks and environments	48	2.00	4.00	3.4375	.61562
Is the design driven and refined by user-centered evaluation in IUP and CASI process	47	2.00	5.00	3.4043	.90071
Are users involved throughout the design and development process in AUSE lifecycle	48	2.00	5.00	3.7917	.87418
The processes in AUSL are iterative	48	1.00	5.00	3.6250	.89025
Is design phase (IUP and CASI) addressing the whole user experience	48	1.00	5.00	3.4167	1.10768
Is the design team included multidisciplinary skills and perspectives	48	1.00	5.00	3.6458	1.06170
Valid N (listwise)	47				

System Requirements Analysis Process

System Requirements Analysis Process is to convert the described stakeholder requirements into a set of preferred system specialized requirements that will monitor the style of system. The reliability score of this section checked by SPSS was 0.7579.

Usability Standards

The standard describes 6 key principles of human centered design act as a manifesto for the field of user experience. This process standard is responsible for managing design processes and gives an overview of the activities that are recommended for human centered design. The reliability score of this section is 0.829.

5 Conclusion

Agile methods were designed to develop software in rapid nature. Though it was successful to produce software but it's a working software not usable software. From the literature review it is obvious that to develop a usable software the agile methods need to incorporate usability approaches. In addition understanding of your end-user is important in order to get usability of your software. This paper has produced a variety of contributions: literature review, survey, experiments and results. From the literature and proposed life cycle we derived that there are many benefits that can be achieved by integrating usability in the agile software model. A few major loopholes were succinctly explained under the heading of current gaps in software models. A survey was conducted among IT professionals to analyze Usability Evaluation in agile software development. After getting the survey results, a proposed agile mode i.e. Agile Usability Software Engineering Lifecycle is proposed. Meanwhile the AUSE life cycle was validated following the IEEE Std 12207-2008 and ISO 9241-210 (Usability standards).

References

1. Kikuno: Why do software projects fail? Reasons and a solution using a Bayesian classifier to predict potential risk. In: 11th IEEE Pacific Rim International Symposium (2005)
2. Masood Butt, S., Ahmad, W.F.W.: Handling requirements using FlexREQ model. In: 2012 IEEE 3rd International Conference on Software Engineering and Service Science (ICSESS), pp. 661–664. IEEE (June 2012)
3. Butt, S.M., Ahmad, W.F.W.: Analysis and Evaluation of Cognitive Behavior in Software Interfaces using an Expert System. International Journal 5 (2012)
4. Butt, S.M., Ahmad, W.F.W., Fatimah, W.: An Overview of Software Models with Regard to the Users Involvement. International Journal of Computer Science Issues (IJCSI) 9(3(1)), 107–112 (2012)
5. Ollson, E.: What active users and designers contribute in the design process. Interacting with Computers 16, 377–400 (2004), `http://www.elsevierComputerScience.com`
6. Koskela, A.: Software configuration management in agile methods. ESPOO, p. I-54. VTT publication 514 (2003)

7. Lee, J.C., McCrickard, D.S.: Towards extreme (ly) usable software: Exploring tensions between usability and agile software development. In: Agile Conference (AGILE), pp. 59–71. IEEE (August 2007)
8. Kane, D.: Finding a Place for discount usability engineering in agile development: Throwing down the gauntlet. In: Proc. Agile Development Conference (ADC 2003), p. AO-46. IEEE Press (2003)
9. Fox, D., Sillito, J., Maurer, F.: Agile methods and user-centered design: How These Two methodologies are being successfully integrated in industry. In: Proc. AGILE 2008 Conference (Agile 2008), pp. 63–72. IEEE Press (2008)
10. Sy, D.: Adapting usability investigations for agile user-centered design. Journal of Usability Studies 2(3), 112–132 (2007)
11. Sohaib, O., Khan, K.: Integrating usability engineering and agile software development: A literature review. In: 2010 International Conference on Computer Design and Applications (ICCDA), vol. 2, p. V2-32. IEEE (June 2010)
12. Najafi, M., Toyoshiba, L.: Two case studies of user experience design and agile development. In: Proc. AGILE 2008 Conference (Agile 2008), pp. 2167–2177. IEEE Press (2008)
13. Sharp, H., Robinson, H., Segal, J.: Integrating user centered design and software engineering: a role for extreme programming, `http://www.ics.heacademy.ac.Uk/events/presentationsl376_hcie` (accessed: December 2013)
14. Memmel, T.: Agile human-centered software engineering. In: Proc. HCI 2007, pp. 167–175. British Computer Society Press (2007)

Emulating Pencil Sketches from 2D Images

Azhan Ahmad, Somnuk Phon-Amnuaisuk, and Peter D. Shannon

Media Informatics Special Interest Group,
School of Computing and Informatics,
Institut Teknologi Brunei,
Mukim Gadong A, BE1410, Brunei Darussalam
{azhan.ahmad,somnuk.phonamnuaisuk,peter.shannon}@itb.edu.bn

Abstract. In this paper we present a pixel-based approach to the production of pencil sketch style images. Input pixels are mapped, using their intensity via a texture-map, to the output sketches. Conceptually, pixels are grouped into regions and the texture obtained from the Texture-map is applied to the output image for a given region. The hatchings and cross-hatchings textures give the resultant images the likeness of pencil sketches. By altering the texture-map applied during the transformation, good results can be obtained, often closely mimicking human sketches. We present details of our approach and give example of sketches. In future work, we wish to enrich the texture-maps so that the texture could better reflect or hint the surface properties of objects in the scene (e.g., hardness, softness, etc.).

Keywords: Emulating pencil sketches, Digital art, Image processing.

1 Background

Computer graphics researchers have, for decades, strode to achieve more life-like results which normally, although not universally, equate to photographs. There is clearly a huge demand on techniques and systems which can pre-render highly sophisticated scenes almost indistinguishable from reality for cinema, television and real-time computer games. Another broad swathe of research, Non-Photorealistic Rendering (NPR), attempts not to emulate the real world but is inspired by artistic styles and conventions of diverse disciplines such as: painting, drawing, technical illustration, comics and animated cartoons. Emphasising style over photorealism can, dependent on context, confer advantages: abstract representation may be more appropriate as elements of heavily styled works, such as, magazine layouts; technical illustration and sketching representations may help focusing on the important elements and messages in a scene.

In this paper, we present our pixel-based pencil sketching emulation approach, which is related to the second category, NPR. Our approach transforms colour and greyscale 2D images into 2D pencil sketches without knowledge of the objects in the scene. Following the convention of more technical drawing styles, texture and tone in the output are rendered as hatchings (or crosshatchings). These hatchings are defined

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_54, © Springer International Publishing Switzerland 2014

in an ordered series, from light to dark. The intensity of a given pixel determines which hatching, from the tone-maps, is rendered in the output. The paper is organised into the following sections. Section 2 gives an overview of related works. Section 3 discusses our proposed concept and gives the details of the techniques behind it. Section 4 provides sample output of the proposed approach. Finally, the conclusion and further research are presented in section 5.

2 Literature Review

Our visual systems interpret a 2D object projected on to the retina as a 3D object in a 3D space. Our brains are trained and have become specialised in recognizing 3D shapes from their 2D projections. The shade and tone give the visual system all the clues needed to perform this visual interpretation. The visual system is so effective and robust that an image with imperfect shade and tone is still correctly perceived as a 3D object. Sketches, therefore, can be drawn in different artistic styles, ranging from realistic to non-realistic. We will classify the main algorithmic approaches in generating 2D sketches as either: pixel-based, model-based or physical model-based. Interested readers can read more background information of this research area in [1, 2].

A pixel-based approach commonly manipulates pixels at the local level. Global information about the objects in the image is not usually available to the creative process. Local pixel information has been employed to generate strokes in [3]. Pixel information is also used to control hatching texture-map or tone-map [4]. Pixel-based techniques have been successfully implemented as various kinds of filters. The 2D sketches can be obtained by combining output from different filters [5, 6]. The concept of artistic filters has been implemented in commercial products such as Photoshop. The tool allows users to interactively select appropriate filters and tonal adjustments to emulate the desired style of pen or pencil strokes. Pixel-based approach like this are usually quite simple and produces illustration that lacks in aesthetic qualia compared to a model-based approach.

A model-based approach commonly uses a 3D model, edge and shape detection algorithms or human expertise during the creative process. The extra information can be included in the rendering process identifying and modifying attributes of a brush stroke used to render the sketch, such as, direction, pressure and brush shape. The availability of the 3D models allows global information to be included in the creative process of a resultant 2D sketch [7-9]. The model-based approach offers a rich platform and has been employed extensively by the Non-Photorealistic Rendering (NPR) community, and while it generally produces greater quality of output compared to a pixel-based approach, the processes involved are, however, complex. For example, Lewis et. al, presented a variation of simulating pencil sketch by segmenting the strokes via silhouette tracing of a 3D models [10].

A physical model-based approach attempts to model the physic of graphite, ink and their interactions on different kinds of paper [11, 12]. This enables the design of interactive drawing tools to be more realistic. Some researchers focus on the design of brush strokes to facilitate the drawing process and do not focus on the physical modelling of brush and how the ink interacts with the paper [13].

There are also many works that focus on creating various brush styles for interactive drawing purposes. This may be seen as remotely related to the physical model-based approach since researchers focus on the appearance of strokes. For example, the Orient-able texture system by Salisbury et. al. [14, 15] is an interactive application which employs the concept of simulating pen-and-ink strokes by positioning the directions and placements of strokes within an illustration. The system depends on three components to produce the final output; a greyscale original image which defines the tone of the illustration, a direction field which sets the orientation of strokes and finally a stroke set which consists of the different strokes that will be used on the illustration. The rendering process employed in this system involves making a continuous blurring of the output illustration and comparing it with the greyscale image until a specific threshold is met which decides on how close the current output is to the greyscale image. The rendering will ultimately stop when the illustration is close to the threshold, producing the final rendering outcome.

3 Our Approach

Given an image *I*, where *I(u, v)* denotes the pixel information at position *(u, v)* of the image *I*. We can denote a spatial domain process applied to pixels in *I* and produce result in *R* using the expression:

$$R(i,j) = T[I(u,v)] \ . \tag{1}$$

Therefore the pixel *R(i, j)* is the result of the transform function *T* defined over a specified neighborhood about point *I(u, v).* A pixel-based approach emulates a pencil sketch by applying a series of appropriate transform functions to an input image. Let us go through some formal definitions of concepts used in our approach below:

Definition 1. *Intensity transformation function: Let I be an input image and Let G be a grayscale image where G(i, j) = T[I(u, v)] where T is the intensity transformation function which can be formally described as a convolution process*:

$$G(i,j) = \sum_{u,v} H_{i-u,j-v} I(u,v); \ \ where\ H\ is\ a\ 3 \times 3\ kernel\ \ . \tag{2}$$

Definition 2. *Mapping between texture-maps and intensity: Let M be a set of user defined texture-maps*[1] *where $M_k(i, j)$ is the intensity of the pixel at (i, j) of the Texture-map k. Let $F_m : R \rightarrow R$ be a function that map G(i, j) to $M_k(i, j)$ and let $F_s : R^2 \rightarrow R$ be a function that map G(i, j) and $M_k(i, j)$ to the final sketch S(i, j) according to some user defined preferences.*

[1] See Examples in Figure 1.

$$M_k(i,j) = F_m(G(i,j)) \ . \tag{3}$$

$$S(i,j) = F_s(G(i,j), M_k(i,j)) \ . \tag{4}$$

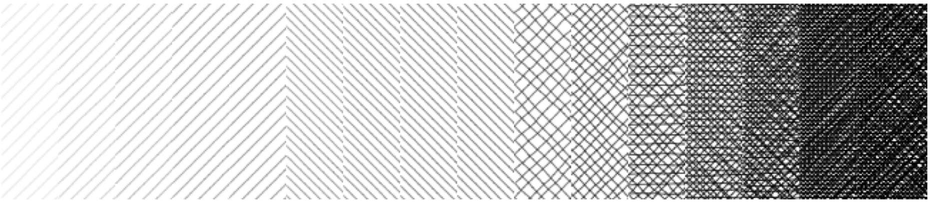


Fig. 1. Examples of texture-maps created using hatching and cross hatching

3.1 Emulating Pixel-Based Pencil Sketches

The main component of our sketching emulation is texture-maps and the intensity information of pixels. For a given input image, the intensity of each pixel is mapped to an output texture using predefined texture-maps (see Figure 1).

1. Perform intensity transformation on the input image *I* (a colour image will be converted into a grayscale image): This process employs a 3×3 kernel *H* (see Eq. 2), for example:

$$H = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix} \text{ or } \begin{bmatrix} -1 & 0 & 2 \\ -1 & 0 & 2 \\ -1 & 0 & 2 \end{bmatrix}$$

   Depending on the types of kernel, various qualities can be produced in the output image, for example, the first kernel in the above example will sharpen the image, this will produce a clearer and sharper contour than the second kernel which place more emphasize on contrast in the vertical direction.
2. Generate different texture-maps of the same size as an input image: the texture-maps provide the desired texture and tones to the generated sketches. Figure 2 shows examples of effects from different texture-maps.
3. Inspect each pixel in *G* where *G(i, j) = T[I(u, v)]* and map it to the desired Texture-maps i.e., $M_k(i, j) = F_m(G(i, j))$.
4. Generate different texture-maps of the same size as an input image: the texture-maps provide the desired texture and tones to the generated sketches. Figure 2 shows examples of effects from different texture-maps.

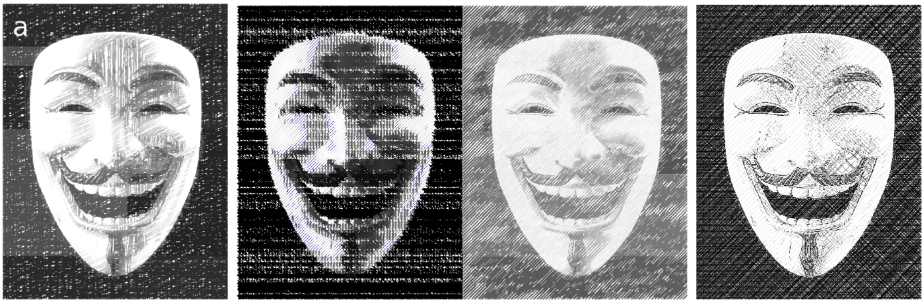


Fig. 2. Examples of sketches generated using different texture-maps

5. Inspect each pixel in G where $G(i, j) = T[I(u, v)]$ and map it to the desired Texture-maps i.e., $M_k(i, j) = F_m(G(i, j))$.
6. Inspect each pixel in G and M and modifies the pixel in the sketch S, i.e., $S(i, j) = F_s(G(i, j), M_k(i, j))$. The function F_s perform the following transformations:

$$\forall i, j M_k(i, j) < tr \Rightarrow S(i, j) = G(i, j) \ .$$

$$\forall i, j M_k(i, j) \geq tr \Rightarrow S(i, j) = 255 \ .$$

where tr = 250 is a users' defined threshold (i.e., to distinguish between strokes and white paper of the map M_k); 255 means white paper.

4 Sketches Emulations and Discussion

The main mechanism in our approach can be seen as superimposing parts of different texture-maps together to form a 2D sketch. In the current implementation, desired texture-maps are selected based on pixel intensity. Figures 3 shows pencil sketches output from the system, original images are also included for readers' convenient in Figure 4. All images used in this work are prepared by the authors (except the Guy Fawkes mask[2]). The image of Elvis has been caricaturised using the process described in [17].

It is always a challenge for researchers in this area to argue the merits of their works. There are no standard benchmark criteria and the evaluation is something of a Turing test. That is, it is a subjective judgment that the system can produce convincing sketches. It is, of course, incumbent on us to attempt to assess at least which elements in our examples work and which are a little underwhelming.

The contrast in intensity of areas within an image affects the quality of the transformation. Images composed of regions populated with pixels whose tonal range is similar to its neighbouring regions tend to produce much more pleasing results e.g., Figure 3 *a*, *b* and *c*. Interestingly, Figure 3 *b* which passes our notional modified Turing Test well, it appears to be hand drawn and suffers from a serious loss of detail, not an uncommon issue with hand drawn sketches but not a desirable trait.

[2] Retrieved from `www.picstopin.com`

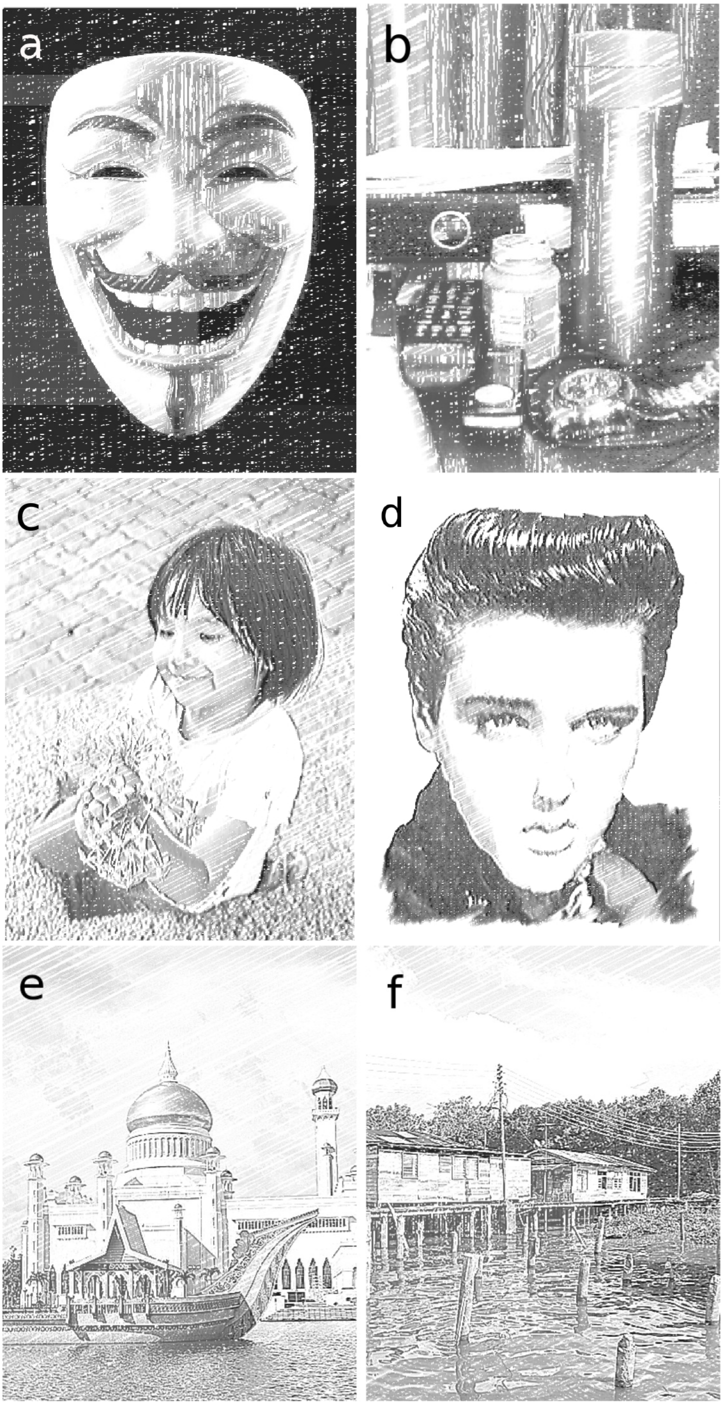


Fig. 3. Sketches of sceneries and portraits

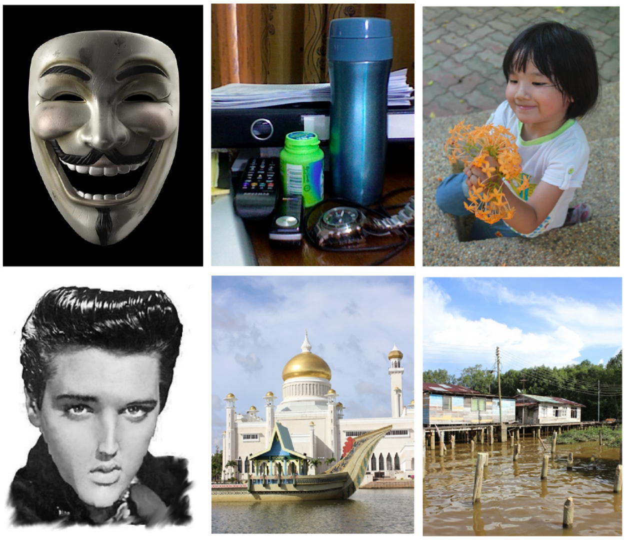


Fig. 4. Original images of sceneries and portraits: Top row from left to right: a, b, and c; Bottom row from left to right d, e, and f)

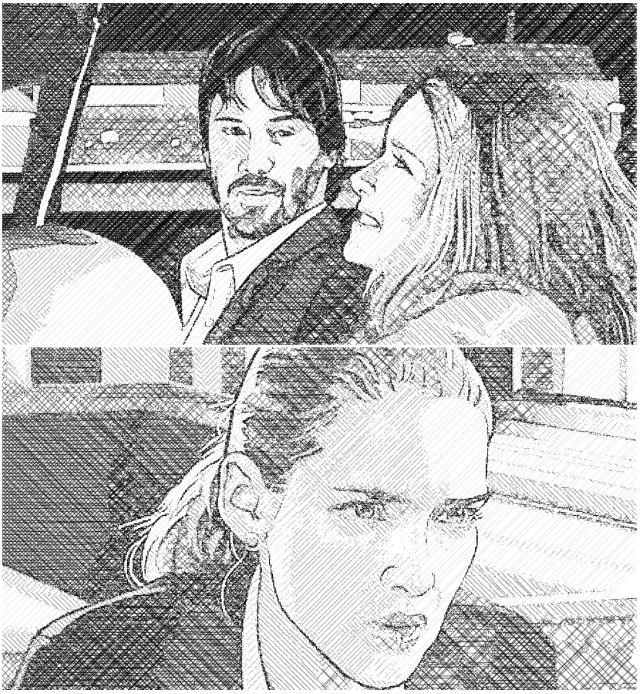


Fig. 5. Sketches of scenes captured from the movie Scanner Darkly

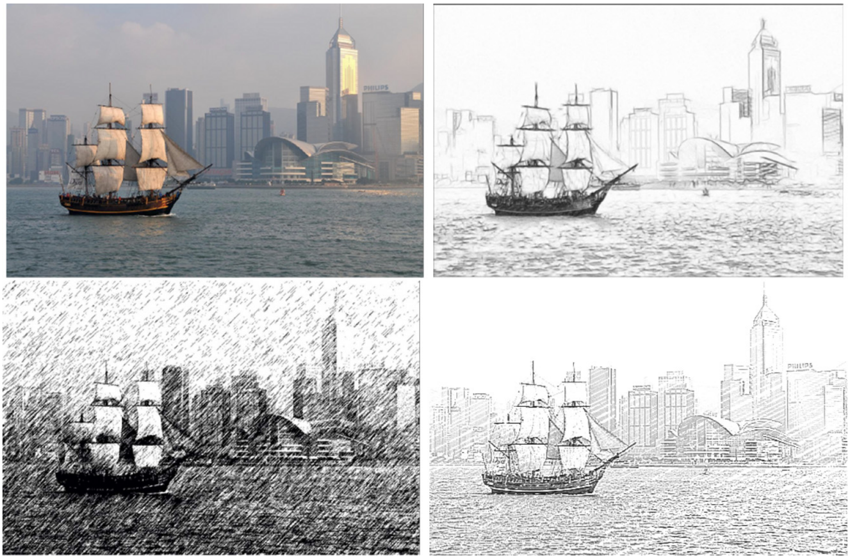


Fig. 6. Comparing output from our system to output from [18] and Photoshop. Clockwise from Top-left: original image, output from [18], output from our system and output from Photoshop.

Turning to photographs in more natural settings, the results are more variable. This seems to be linked to the problem of poor contrast, noted above, presents. In Figure 3 *e* both the skies and the bright walls of the mosque and translate very pleasingly. The more detailed and noisy sections, like the water and trees, look very much like black and white photocopies and generally unsatisfactory.

The images in Figure 5 are, perhaps, the most successful of those included here. They are taken from the cult film, named after the novel by Philip K. Dick, A Scanner Darkly. A multitude of visual styles were used in this Film; the final frames used were created using a technique called Interpolated Rotoscoping. Although automated in this case, when first used each frame would be painted over by hand. The final result is similar to a moving comic book. Using these stylised images for input, the result are very good and leads us to suggest a second feature of images which are amenable to transformation: dramatic changes in intensity of regions are successfully transformed if they are both well-defined and the regions are large enough. This affect can be noted in Figure 3 *a*, in the sharply defined edge of the mask against the black background. Finally, we compare our output image to different images produced from [18] and from the popular commercial package, Photoshop artistic sketch filter (see Figure 6).

We believe that this approach produces a good output in emulating pencil sketches and has great potential for automating creation of computer-generated images in many types of printed material for certain classes of drawings, as defined above. This approach employs a relatively simple two-layer process; applying transformation filter to an image and mapping pixels to a texture-map; without intervention of a user and is certainly fast enough be used to transform video in real-time.

5 Conclusion and Future Work

We have shown that the predefined Texture-maps can be used to effectively transform an image to give the impression of a hand drawn sketch. The transformation is computationally inexpensive, regardless of the number of texture-maps employed in the process. This approach provides a quick sketch since the mapping between the input pixels and the Texture-maps can be done just in one pass.

Much of the information artists impart to those who see their work is through two basic techniques: firstly the use of pressure and density of pencil strokes to suggest shadow and highlight; and, secondly, the path of the pencil strokes, relative to each other, when shading to create the illusion of shape, depth and perspective. The process described in this paper is akin to the first technique; tonal information is used to select an appropriate texture for the transformation, without 3D model information to control the stroke directions.

We also identified that the contrast on the input image influenced the output. To further explore the effectiveness of this system, the input image will need to be processed to get a good contrast before mapping the Texture-maps. In future work, we hope to explore this technique further by providing a context sensitive Texture-maps i.e., Texture-maps that exhibit properties of the 3D objects. For example, using curve stroke to signify curvature in the 3D objects; using different stroke qualities to signify different physical properties (e.g., hardness, softness, etc). Another avenue that we can explore is the post-rendering of video footage and live feedbacks.

References

1. Strothotte, T., Schlechtweg, S.: Non-Photorealistic Computer Graphics: Modeling, Rendering and Animation. Morgan Kaufmann Publishers, Elsevier Science, USA (2002)
2. Hertzmann, A.: A survey of stroke-based rendering. IEEE Computer Graphics and Applications 23(4), 70–81 (2003)
3. Haeberli, P.E.: Paint by numbers: Abstract image representation. In: Proceedings of the 17th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1990. Computer Graphics Proceedings, vol. 24, pp. 207–214. ACM Press (1990)
4. Yang, H., Kwon, Y., Min, K.: A texture-based approach for hatching color photographs. In: Bebis, G., et al. (eds.) ISVC 2010, Part I. LNCS, vol. 6453, pp. 86–95. Springer, Heidelberg (2010)
5. DeCarlo, D., Santella, A.: Stylization and abstraction of photographs. In: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2002, pp. 769–776. ACM Press (2002)
6. Kasao, A., Miyata, K.: Algorithmic painter: A NPR method to generate various styles of painting. Visual Computing 22, 14–27 (2005, 2006)
7. Winkenbach, G., Salesin, D.H.: Rendering parametric surfaces in pen and ink. In: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1996, New Orleans. Computer Graphics Proceedings, Annual Conference Series, pp. 469–476. ACM SIGGRAPH, New York (1996)
8. Isenberg, N., Halper, T., Strothotte, T.: Stylizing silhouettes at interactive rates: From silhouette edges to silhouettee strokes. Computer Graphics Forum 21(3), 249–258 (2002)

9. Wilson, B., Ma, K.L.: Rendering complexity in computer-generated pen-and-ink illustrations In. In: Proceedings of the 3rd International Symposium on Non-photorealistic Animation and Rendering (NPAR 2004), pp. 129–137 (2004)
10. Lewis, J.P., Fong, N., Xiang, X.X., Soon, S.H., Feng, T.: More Optimal Strokes for NPR Sketching. In: Proceedings of the 3rd International Conference on Computer Graphics and Interactive Techniques in Australia and South East Asia, pp. 47–50 (2005)
11. Sousa, M.C., Buchanan, J.W.: Observational model of graphite pencil materials. Computer Graphics Forum 19(1), 27–49 (2000)
12. Zhang, Q., Sato, Y., Takahashi, J., Muraoka, K., Chiba, N.: Simple cellular automata-based simulation of ink behaviour and its application to suibokuga-like 3D rendering of trees. Journal of Visualization and Computer Animation 10(1), 27–37 (1999)
13. Chu, N.S.H., Tai, C.L.: An efficient brush model for physically-based 3D painting. In: Proceedings of the 3rd International Conference on Computer Graphics and Interactive Techniques in Australia and South East Asia, pp. 47–50 (2005)
14. Salisbury, M.P., Anderson, S.E., Barzel, R., Salesin, D.H.: Interactive pen-and-ink illustration. In: Proceedings of the 10th Pacific Conference on Computer Graphics and Applications, pp. 413–421. IEEE Press (2002)
15. Salisbury, M.P.: andWong, M., and Hughes, J., and Salesin, D.: Orientable textures for image-based pen-and-ink illustration. In: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1997. Computer Graphics Proceedings, pp. 401–406. ACM Press (1997)
16. Sauvaget, C., Boyer, V.: Comics stylization from photographs. In: Bebis, G., et al. (eds.) ISVC 2008, Part I. LNCS, vol. 5358, pp. 1125–1134. Springer, Heidelberg (2008)
17. Phon-Amnuaisuk, S.: Exploring particle-based caricature generations. In: Abd Manaf, A., Zeki, A., Zamani, M., Chuprat, S., El-Qawasmeh, E. (eds.) ICIEIS 2011, Part II. CCIS, vol. 252, pp. 37–46. Springer, Heidelberg (2011)
18. Lu, C.W., Xu, L., Jia, J.Y.: Combining sketch and tone for pencil drawing production. In: Proceedings of the Symposium of Non-Photorealistic Animation and Rendering (NPAR 2012), pp. 65–73. Eurographics Association Aire-la-Ville, Switzerland (2012)

Router Redundancy with Enhanced VRRP for Intelligent Message Routing

Haja Mohd Saleem[1], Mohd Fadzil Hassan[2], and Seyed M. Buhari[3]

[1] Institut Teknologi Brunei, Mukim Gadong A, BE1410, Brunei Darussalam
mohamed.saleem@itb.edu.bn
[2] Universiti Teknologi PETRONAS, Malaysia
mfadzil_hassan@petronas.com.my
[3] King Abdul Aziz University, Kingdom of Saudi Arabia
mibuhari@yahoo.com

Abstract. Overlay query routing mechanism is a popular approach for query routing process in the distributed service and resource discovery. However it suffers from drawbacks such as escalated inter-ISP traffic and redundant traffic forwarding in the underlying IP layer. In order to avoid these problems we have proposed earlier that the overlay query routing process could be moved down to the IP layer with the help of intelligent message routing (IMR). The routers in the IP layer build a second routing table by mapping the content of the query messages with the target location of the services which is used for query forwarding. For such a system to be implemented in the Internet scale, high availability of routing service is vital. Employing Virtual Router Redundancy Protocol (VRRP) for redundancy takes care of classical route updates to the backup router. However, the service specific routing table which is specific for underlay query processing needs to be updated independent of VRRP. In this paper, we address the issue of applying router redundancy for IMR with enhanced VRRP which also can handle the redundancy for the service specific routing table. We have also analyzed the performance of routers with respect to the additional overhead taken due to service specific routing.

Keywords: Intelligent Message Routing, VRRP, AON, Service Discovery.

1 Introduction

To handle the drastic growth in the usage of Internet effectively, certain processes such as load balancing and message-aware forwarding that are performed by end nodes ought to be performed by various networking components like routers and switches [1]. As such, Next Generation Internet needs IMR to be a part of the core routing process [2]. This growth in demand impacts the network environment with inefficient route selection, network congestion, network performance degradation, ISP-based traffic management exhaustion, etc.

CISCO has responded to this need with Application Oriented Networking (AON). AON, with its capability of intelligent message routing and switching, can

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_55, © Springer International Publishing Switzerland 2014

differentiate between a normal and AON specific packet [3]. AON packets are processed based on its message contents and routed/switched using the AON module. One such application of AON is IMR in the service discovery domain [4].

Much of the present search heuristics for service discovery algorithms function at the overlay layer of the service discovery systems. They do not possess knowledge of the target location in the underlying network which could leverage the performance if exploited suitably. Queries generated during service discovery can be routed based on their contents using AON. This approach of routing based on the message contents is termed as IMR. For instance, a query, searching for a lowest airfare for a particular destination will be forwarded to the service registries related to airlines sectors. In this case the query routing is performed in the IP-layer instead of the overlay layer that is usually formed by service discovery applications. By performing the query routing in the IP-layer, the performance of the system improves by reducing the inter-ISP traffic and redundant query forwarding in the underlying network topology. For such a system, the overall application performance depends on the high availability of networking nodes in the IP-layer. Applying VRRP as such in the IP-layer provides redundancy only to the classic routing tables. So far, to the best of our knowledge there has not been any attempt to provide redundancy for IMR. This issue needs to be addressed by providing redundant routers that runs the VRRP protocol which also supports redundancy for IMR. In this paper we provide redundancy for IMR and also evaluate its feasibility through simulation. The rest of the paper is organized as follows. Section 2 discusses related work, section 3 explains the application of AON is service discovery process, section 4 discusses the methodology and section 5 details the implementation and experimental setup. The results obtained through simulation are discussed in section 6 and section 7 concludes the paper.

2 Related Work

Service discovery has been addressed by many authors in the recent literature. Finding the location information in the underlay could be performed using either dynamic or a static approach. PIPPON [5] clusters peers that are with a specific proximity and tries to bring the underlay awareness to the overlay. The proximity of peers is based on Longest Prefix IP Matching and Round Trip Time (RTT), thus making location identification dynamic. But, the authors did not consider the similarity of queries in service discovery. TOPLUS [6] clusters peers based on three-tier hierarchy, namely, groups, super-groups and hyper-groups. Proximity identification is done using XOR metric. Also location identification is dynamic in nature. TOPLUS does not consider inter-ISP and redundant traffic along with interlayer communication overhead. Plethora [7] uses two-layer architecture: cSpace and gSpace. cSpace is the local broker while gSpace is the global broker. To identify the location of the service, static approach is applied here. Similar to Plethora, P4P [8] uses iTrackers and appTrackers to address the locality awareness problem. iTrackers are present at every ISP but appTrackers reside globally. appTrackers are able to have a complete picture of the P2P applications.

It is relevant to note that IETF has taken the initiative to optimize the traffic in the underlay by forming Application Layer Traffic Optimization (ALTO) group in 2009. Seedorf et al. [9] uses the service of ALTO group to P2P applications. Therefore, P2P peers could obtain the topological and proximity information from the ALTO server. This information could be used to cause security issues by any malicious user. SLUP [10] forms clusters based on semantics and RTT. Clusters like normal peer, level-2 super peer and level-1 super peer are formed but redundancy in the underlay is not considered. Bindal et al. [11] has applied proximity information in the case of BitTorrent network in order to reduce the inter-ISP traffic. By considering neighbours from the same ISP, inter-ISP traffic is reduced. Shen et al. [12] has proposed caching of P2P traffic similar to HTTP traffic caching by ISPs. Service discovery is not considered in the work. P4P Pastry [8] discusses mechanisms to bring locality features to the PASTRY [13] structured system.

Certain works have already been done in the area of AON. For instance, Yu Cheng et al. [14] have designed a model using AON for automatic service composition. Combination of Service Oriented Architecture (SOA), AON and automatic service computing is performed to solve scalability and performance issues. Tian et al. [15] have enhanced AON into “Application Oriented Studies”. Authors have proposed the use of a specific multicast approach to replace the overlay based multicast. Yao et al. [1] have used AON-based switching to perform effective message scheduling in datacenters. Kamal et al. [16] have discussed about assisting AON with Service Location Problem (SLP) to improve the efficiency of the system. Authors in [17] have studied the impact of path utilization and bandwidth fairness when different techniques are used to multipath connectivity. Authors have also compared various load balancing and path forming algorithms with or without redundancy. It has been concluded that there is no ideal algorithm for load balancing between multiple, unequal paths. Jing Fu et al. [18] have compared the efficiency of centralized routing scheme with that of the link-state decentralized routing scheme. It argued that centralized routing scheme performs faster compared to decentralized approach. The problem of single point of failure could be addressed using protocols like VRRP.

The implementation of AON in the domain of service discovery has been published in our earlier works [4,19-21] which is capable of mapping the service classes to the geographical location on the target registries. This paper particularly caters to fault tolerance for Distributed Service Discovery (DSD) by providing redundant routers that operate using VRRP.

3 Methodology

In our earlier works [4, 19-21] we have elaborated an implementation of IMR with the help of AON in the domain of DSD. We introduced an IP sub-layer which is capable of performing IMR based on the content of search queries generated by DSD processes. The algorithm used for IMR routing is shown in Figure 1. As it can be seen from the algorithm the AON router plays a vital role in the success of the DSD process. The DSD process is not left to the application layer alone. It functions in the application layer at the time of query generation and at the time of receiving the query. The intermediate processes and forwarding decisions are made by the AON routers. Therefore, the failure of an AON router is likely to cause the failure of DSD process itself.

The technology for providing router redundancy already exists in the form of VRRP protocol. The functioning of VRRP protocol is shown in Figure 2. Once the VRRP is configured for the master and backup routers on a network, one of the IP addresses becomes the default gateway of the VRRP group. As shown in Figure 2, 10.0.0.1 is assigned to the master router. If the master router fails, the same IP address 10.0.0.1 will point to the backup router which is automatically handled by VRRP. VRRP does this by means of virtual media access control (MAC) address which is reassigned to the backup router once the master fails.

```
Algorithm: Routing
Input: packet
IF packet = AON THEN
  IF packet = DSD THEN
     IF packet = request THEN
       IF AON routing table entry exist THEN
           Route as per the entry
       ELSE
           Forward to all egress ports
     ELSE
       Update the AON routing table
       Forward the packet in the return path
  ELSE
rward to the appropriate AON module
ELSE
     Perform classical routing
```

Fig. 1. Query routing algorithm

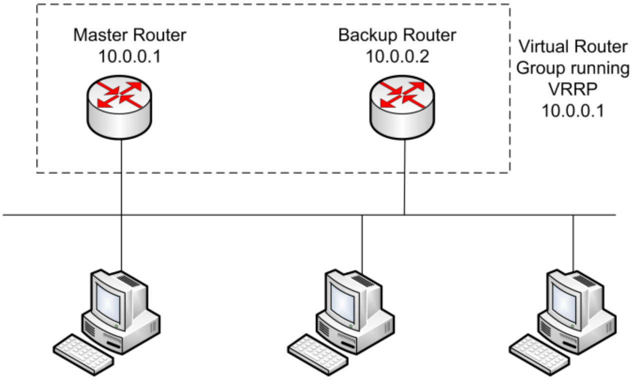


Fig. 2. Virtual Router Redundancy Protocol (VRRP)

In our scenario, after implementing VRRP the service registries are connected with redundant routers as shown in Figure 3. The AON Routing Table (AON RT) needs to be updated to the backup router since VRRP based normal IP layer route update does not include AON routing updates.

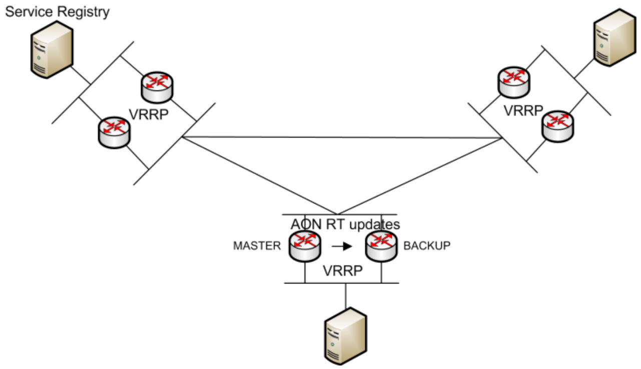


Fig. 3. AON Routing updates for DSD

4 Experimental Setup and Discussion

A topology consisting of ten service registries and fourteen VRRP block routers has been set up as shown in Figure 4. The registries are classified based on the type of services they offer. This search query is also constructed with the type of the query message they are looking for. The IMR algorithm matches the query with the location of the registries and forwards them accordingly in the underlay. Further details of this process can be obtained from our previous work [4, 19-21].

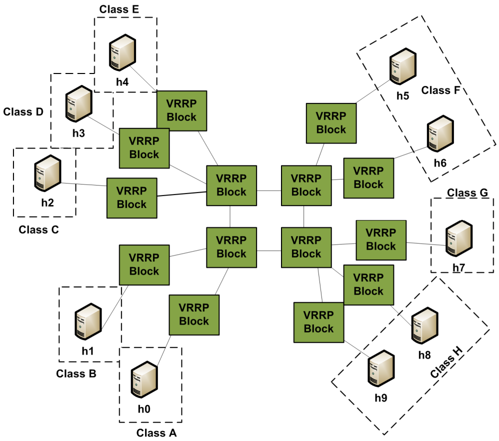


Fig. 4. Experimental Setup

Various search queries corresponding to the service discovery process are generated. The search query format is shown in Figure 5.

Request/ Reply Integer: 0/1	Time stamp, float	Query message, String	Result, String	Is AON, Boolean	Previous node, String	Hop count, Integer	Query ID

Fig. 5. AON Query Format

Queries contain the following fields:

- Request/Reply field is used to indicate whether the message is a request or reply message
- Timestamp field is used to calculate the RTT value
- Query message field is used to indicate the actual query and the service class information that is required by the AON router
- Result field is used to store the result of the query
- isAON flag is used to identify whether a packet is AON packet or not. This field could be used in the IPv6 extension headers
- Previous node field is used to indicate the last arrived node on the path of the packet
- Hop count is used to identify the number of hops crossed by the query
- Query ID field is used to identify the query and prevent it from looping

In order to compare the performance between the system without VRRP and the one with VRRP, different scenarios are created. These scenarios are designed considering the time interval for performing the routing updates to the backup router. Updating the backup router frequently keeps it up-to-date but cause excess network traffic. However, delaying the update is likely to keep the backup router information out-of-date. In case the primary router goes off in this status, backup router won't have the latest routing information. Thus, there is a tradeoff between performance and efficiency of the system.

```
Algorithm: Routing
Input: packet
IF packet = AON THEN
  IF packet = DSD THEN
     IF packet = request THEN
       IF AON routing table entry exist THEN
           Route as per the entry
       ELSE
           Forward to all egress ports
     ELSE
      Update the AON routing table in the main router
      Update the AON routing table in the backup router
      Forward the packet in the return path
ELSE
Forward to the appropriate AON module
ELSE
  Perform classical routing
```

Fig. 6. Modified algorithm from Figure 1

The different scenarios used for our testing were:

1. Normal AON routing process as per the algorithm given in Figure 1
2. Immediate update of backup router whenever there is an AON RT update in the main router
3. Backup router routing table update with 1 millisecond interval (background process)
4. Backup router routing table update with 1000 milliseconds interval (background process).

The modified AON routing algorithm is shown in Figure 6 and 7. Figure 6 corresponds to the second scenario given above whereas Figure 7 corresponds to the 3rd and 4th scenarios.

```
Algorithm: UpdateBackup

Run update thread at a regular preset interval

Update the AON routing table in the backup router
```

Fig. 7. Router update algorithm with a backup thread

5 Results and Observation

The experimental setup shown in Figure 4 is simulated using J-Sim 1.3 and evaluated for various scenarios. Random search query requests corresponding to DSD are generated to test the network. This random set of query requests is then applied to the network with and without VRRP support. Query response time for various scenarios are collected and compared. Redundancy updates in VRRP is done either as part of the same thread that does AON update or in a separate thread. The different scenarios that emerge are as follows,

- Upon changes on the main AON router, updates are made to the backup router immediately using the same thread
- Updates on main AON router are updated to the backup router using an independent thread after a time interval of
    - 1ms
    - 10ms
    - 1000ms

In real-time routing updates are not performed as frequent as considered in the test scenarios. The objective for such a close time interval in the scenarios is to study the worst case. As shown in Figure 8, a set of 35 queries is generated to the system with a time interval of 1 millisecond and the query processing time has been observed. The query processing time includes the time to update the router's routing table and also to handle the query itself. The AON router without a backup router consumes the minimum most possible processing time. Using the same thread to update both the main and the backup router causes greater delay in the performance of the router. This is due to the reason that for every update, there is also a need to update the backup router. For certain queries, which do not require new routes, there will be no new route added to the AON routing table. Hence, In this situation routing update is not required. Here the routing update time is zero but due to the query processing time, there is no zero value in Figure 8. A minimum value of around 2.5972 milliseconds is required for the processing of the query along with the required routing update. Maximum time required for query processing is 4.8 milliseconds. Performing the redundant router update in the main thread itself causes a maximum increase of 0.16 milliseconds in query processing time. From the figure it is obvious that the presence of redundant router in AON does not affect the system much and thus redundancy could be provided without much performance degradation.

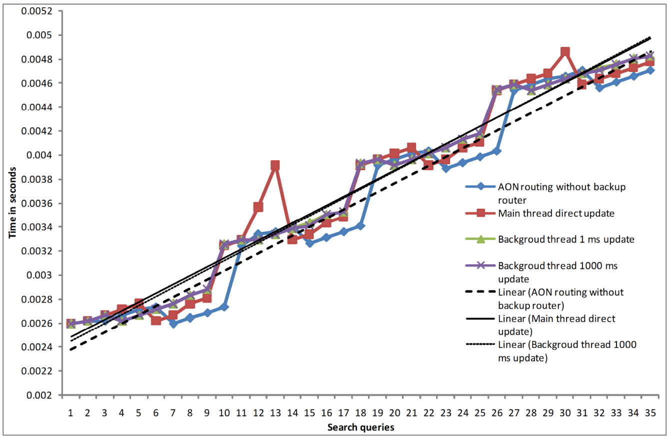


Fig. 8. Search Query Vs Query Processing Time

6 Conclusions

This paper has addressed the issue of providing redundancy using VRRP to the existing AON based service discovery. VRRP takes care of IP-layer routing updates while AON based routing update is performed in parallel. Comparing the AON based service discovery without redundancy, it is clear the performance does not degrade much in providing redundancy which requires regular updates. Future studies may be able to evaluate applying AON update along with Link-state or BGP based routing updates.

References

[1] Yao, J., Ding, J.J., Bhuyan, L.N.: Intelligent Message Scheduling in Application Oriented Networking Systems. Presented at IEEE International Conference on Communications, ICC 2008 (2008)

[2] Chappell, D., Berry, D.: Next-Generation Grid-Enabled SOA: Not Your MOM's Bus. SOA Magazine (2008)

[3] CISCO, Cisco AON: A Network Embedded Intelligent Message Routing System

[4] Saleem, H.M., Hassan, M.F., Asirvadam, V.S.: An Intelligent Query Routing Mechanism for Distributed Service Discovery with IP-layer Awareness. Presented at the International Conference on Informatics Engineering & Information Science (ICIEIS 2011), Universiti Teknologi Malaysia, Malaysia (2011)

[5] Hoang, D.B., Le, H., Simmonds, A.: PIPPON: A Physical Infrastructure-aware Peer-to-Peer Overlay Network. Presented at 2005 IEEE Region 10, TENCON 2005 (2005)

[6] Garcés-Erice, L., Ross, K.W., Biersack, E.W., Felber, P., Urvoy-Keller, G.: Topology-Centric Look-Up Service. In: Stiller, B., Carle, G., Karsten, M., Reichl, P. (eds.) NGC 2003 and ICQT 2003. LNCS, vol. 2816, pp. 58–69. Springer, Heidelberg (2003)

[7] Ferreira, R.A., Grama, A., Jagannathan, S.: Plethora: An Efficient Wide-Area Storage System. In: Bougé, L., Prasanna, V.K. (eds.) HiPC 2004. LNCS, vol. 3296, pp. 252–261. Springer, Heidelberg (2004)

[8] Zhengwei, G., Shuai, Y., Huaipo, Y.: P4P Pastry: A novel P4P-based Pastry routing algorithm in peer to peer network. Presented at 2010 the 2nd IEEE International Conference on Information Management and Engineering, ICIME (2010)

[9] Seedorf, J., Kiesel, S., Stiemerling, M.: Traffic localization for P2P-applications: The ALTO approach. Presented at IEEE Ninth International Conference on Peer-to-Peer Computing, P2P 2009 (2009)

[10] Xin, S., Kan, L., Yushu, L., Yong, T.: SLUP: A Semantic-Based and Location-Aware Unstructured P2P Network. Presented at 10th IEEE International Conference on High Performance Computing and Communications, HPCC 2008 (2008)

[11] Bindal, R., Pei, C., Chan, W., Medved, J., Suwala, G., Bates, T., Zhang, A.: Improving Traffic Locality in BitTorrent via Biased Neighbor Selection. Presented at 26th IEEE International Conference on Distributed Computing Systems, ICDCS 2006 (2006)

[12] Shen, G., Wang, Y., Xiong, Y., Zhao, B.Y., Zhang, Z.-L.: HPTP: Relieving the Tension between ISPs and P2P. Presented at USENIX IPTPS (2007)

[13] Rowstron, A., Druschel, P.: Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems. In: Guerraoui, R. (ed.) Middleware 2001. LNCS, vol. 2218, pp. 329–350. Springer, Heidelberg (2001)

[14] Cheng, Y., Leon-Garcia, A., Foster, I.: Toward an Autonomic Service Management Framework: A Holistic Vision of SOA, AON, and Autonomic Computing. IEEE Communications Magazine (2008)

[15] Xiaohua, T., Yu, C., Kui, R., Bin, L.: Multicast with an Application-Oriented Networking (AON) Approach. Presented at IEEE International Conference on Communications, ICC 2008 (2008)

[16] Zille Huma, K., Ala, A.-F., Ajay, G.: A service location problem with QoS constraints. In: Proceedings of the 2007 International Conference on Wireless Communications and Mobile Computing, Honolulu, Hawaii, USA, pp. 641–646. ACM (2007) 978-1-59593-695-0

[17] Makela, A., et al.: Compairson of load-balancing approaches for multiple connectivity. Computer Networks 56, 2179–2195 (2012)

[18] Fu, J., et al.: Intra-domain routing convergence with centralized control. Computer Networks 53, 2985–2996 (2009)

[19] Saleem, H.M., Hassan, M.F., Asirvadam, V.S.: Proxy-based Selective Forwarding in Distributed Service Discovery using Application Oriented Networking. Advances in Information Sciences and Service Sciences (AISS) 4, 9–18 (2012)

[20] Saleem, H.M., Hassan, M.F., Asirvadam, V.S.: Modelling and Simulation of Underlay aware Distributed Service Discovery. Presented at the 17th Asia Pacific Conference on Communications, Malaysia (2011)

[21] Saleem, H.M., Hassan, M.F., Asirvadam, V.S.: Distributed Service Discovery Architecture: A Bottom-Up Approach with Application Oriented Networking. Presented at the Third International Conference on Emerging Network Intelligence, EMERGING 2011, Lisbon (2011)

Selecting Most Suitable Members for Neural Network Ensemble Rainfall Forecasting Model

Harshani Nagahamulla[1], Uditha Ratnayake[2], and Asanga Ratnaweera[3]

[1] Dept of Computing & Information Systems, Faculty of Applied Sciences,
Wayamba University of Sri Lanka, Kuliyapitiya, Sri Lanka
[2] Institut Teknologi Brunei, Brunei
[3] Dept of Mechanical Engineering, Faculty of Engineering,
University of Peradeniya Peradeniya, Sri Lanka
harshaninag@yahoo.com, uditharr@gmail.com, asangar@pdn.ac.lk

Abstract. Neural network ensembles are more accurate than a single neural network because they have higher generalization ability. To increase the generalization ability the members of the ensemble must be accurate and diverse. This study presents a method for selecting the most suitable members for an ensemble which uses genetic algorithms to minimize the error function of the ensemble ENN-GA. The performance of the proposed method is compared with the performance of two widely used methods, bagging and boosting. The models developed are trained and tested using 41 years rainfall data of Colombo and Katugastota Sri Lanka. The results show that the ENN-GA model is more accurate than Bagging and Boosting models. The best performance for Colombo was obtained by ENN-GA with 14 members with RMSE 7.33 and for Katugastota by ENN-GA with 12 members with RMSE 6.25.

Keywords: Rainfall forecasting, Artificial Neural Networks, Neural Network Ensembles, K-means Clustering, Genetic Algorithms.

1 Introduction

Rainfall forecasts are a necessity in planning many human activities. Decision support systems like flood warning systems, urban water management systems and many others use rainfall forecasts as an input. Hence the accuracy of the forecasts are very important as false predictions can mislead these systems and cause huge problems. Artificial Neural Networks (ANN) are used in forecasting applications because of their ability to handle complicated and imprecise data efficiently. In hydrology ANN are used in many prediction problems like rainfall-runoff prediction, ground-water management and precipitation prediction with acceptable results [1].

The accuracy of a prediction depends on the ANN's generalization ability. A collection of classifiers trained to do the same task is called an ensemble. Hansen and Salamon [2] shows that the generalization ability of an ensemble is significantly increased than of a single classifier, which is also confirmed by our past studies [3, 4, 5]. Many researches are carried out to find methods to further increase the

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_56, © Springer International Publishing Switzerland 2014

generalization ability of the ensembles. Their results show that the generalization ability of an ANN ensemble increases when only a few number of ANN are included in the ANN ensemble rather than all available ANN [6] and also when the member ANN are accurate and diverse [7]. Diversity of ANN is that their errors represents different regions in the input space. Accuracy and diversity are two conflicting conditions that have to be balanced carefully to achieve good performance. The performance of an ensemble also depends on the way the members are connected.

This study presents two methods for selecting the most suitable members for an ANN ensemble. Their performances are compared with the performance of two widely used techniques, bagging and boosting. The rest of this paper is organized as follows. Section 2 provides a review of available literature. Section 3 describes our methodology. Section 4 presents the results obtained from our experiment and provides a discussion and section 5 concludes the paper.

2 Ensemble Techniques

Ensembles are used in both classification and regression applications. There are many examples of using ensembles in forecasting applications. Gheyas and Smith [8] found that an ensemble with a collection of Generalized Regression Neural Networks (GRNN) gives better predictions compared with single GRNN, for time series prediction. Maqsood, Khan and Abraham [9] developed an ensemble model to make 24 hour ahead forecasts for temperature, wind speed and relative humidity which outperform single networks and statistical models.

Creating an ANN ensemble include three steps; creating the set of ANN, selecting the suitable ANN from the trained set and combining the selected ANN to get the output. There are many different methods to perform each of these steps and each step will contribute to the generalization ability and of the ANN ensemble.

2.1 Creating the ANN

The first step of creating an ANN ensemble is creating the member ANN. The result of an ANN can change significantly by a small change in the ANN's various parameters. Sharky [10] explains different techniques to create the members of ANN ensemble by varying the parameters associated with design and training of ANN as follows. Varying the initial random weights while keeping all other parameters the same. Varying the network architecture, like number of layers, number of nodes per layer and the activation functions. Varying the network type like Back Propagation Network (BPN), Radial Basis Function Network (RBFN) and GRNN. Varying the training data. Different training data can be obtained from sampling the available data, using different data sources and using different preprocessing steps.

Training ANN by each of these techniques can create a diverse set of ANN and it was found that varying the training data and network type yield the best results [3, 4, 5]. A pool of ANN with higher diversity can be created by training a number of ANN using a combination of above techniques. In this study the ANN pool will be created by training a set of GRNN by varying the training data. Training data are obtained by using different preprocessing steps and data sampling techniques.

2.2 Selecting the Members

The next step in creating an ANN ensemble is selecting the most suitable members from the trained ANN pool maintaining the balance between accuracy and diversity. There are a number of different methods available for this starting from simple trial and error methods to advanced optimization techniques.

In this study two new models Ensemble Neural Network with K-means clustering (ENN-K) and Ensemble Neural Network with Genetic Algorithms (ENN-GA) will be created. ENN-K uses k-means clustering algorithm to cluster the ANN in the pool according to their error. Errors of ANN in each cluster represent same part of the input space and errors of ANN from different clusters represent different parts of the input space. Hence ANN in each cluster will be diverse from ANN in other clusters.

K-means clustering group n objects into k clusters where each object belongs to the cluster with the nearest mean. If there are n ANN denoted by (x_1, x_2, …, x_n) k-means clustering partition them into k sets where ($k \leq n$) $S = \{S_1, S_2, \ldots, S_k\}$ such that the within cluster sum of squares are minimized using the following equation.

$$\arg\min_{S} \sum_{i=1}^{k} \sum_{x_j \in S_i} \left\| x_j - \mu_i \right\|^2 \tag{1}$$

where μ_i is the mean of points in S_i.

ENN-GA uses a GA to find the most suitable ANN from the pool to be included in the ensemble so that the Mean Squared Error (MSE) of the ensemble is reduced. GA is a search and optimization algorithm that works based on the process of natural selection [11]. GA has a set of solutions to the problem called the population and a set of biologically inspired operators; selection, crossover and mutation. The solutions are evaluated according to an objective function to identify how close a solution is to the expected answer. Only the most appropriate solutions in a population will survive and generate offspring and transmit their characteristics to new generations.

GA has been used for ANN selection for ensembles in previous studies. ADDEMUP [12] creates diverse members using a Genetic Algorithm (GA) rather than selecting from a trained ANN pool. It first creates only an initial population of ANN and using genetic optimisers continually create new networks that are accurate and diverse. GASEN [6] employs genetic algorithm to evolve the weights of ANN to characterize the fitness of the ANN in an ensemble.

2.3 Combining the ANN

The final step is to combine the selected ANN. There are several different methods for combining ANN. Average method assigns the same priority for all ANN and weighted average method assigns a weight to each ANN to minimize the MSE of the ensemble. Majority voting and weighted majority voting methods are two widely used methods in classifier ANN ensembles [13]. Another method that is widely used for combining ANN is called stacking [4, 14] where a separate ANN is used to combine the selected ANN in the ensemble. Evolutionary methods such as GA [15] and intelligent agents [16] are also used in combining ensembles. In this study a GRNN is used to combine the selected ANN and give the final output.

To compare the performance of these models another two models will be created using two widely known techniques, bagging and boosting [17]. Bagging train multiple models on different samples and average their predictions. Boosting algorithm also train multiple models on different samples and assign weights for the models so that the incorrectly predicted samples are emphasized. The models are then combined with weighted average method.

2.4 Measuring Forecasting Accuracy

The measure of how close the forecasted value to the actual value is called the forecasting accuracy. In this study three measurements were used; Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and the Coefficient of Determination R^2. RMSE measures how far the average error is from zero Eq.2. MAE gives an average of the absolute errors Eq.3. Coefficient of determination represents how well the regression line represents the data.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} e_i^2}{n}} \tag{2}$$

$$\text{MAE} = \frac{\sum_{i=1}^{n} |e_i|}{n} \tag{3}$$

Where e is the difference between the actual value and the predicted value.

3 Methodology

3.1 Variable Selection

Colombo located in the western coast of Sri Lanka on North latitude 6° 55' and East longitude 79° 52' was selected as the study area. Colombo is situated in wet zone with an annual rainfall of about 240 cm. Daily rainfall data of Colombo for 41 years (1961-2001) was collected from the Department of Meteorology Sri Lanka as the output of the ANN models.

NCEP_1961-2001 dataset derived from the NCEP reanalysis [18] was taken as the input of the ANN models. The data set contains 26 variables and 41 years (1961 - 2001) of daily observed data. These include Mean sea level pressure, Surface airflow strength, Surface zonal velocity, Surface meridional velocity, Surface velocity, Surface wind direction, Surface divergence, 500 hpa airflow strength, 500 hpa zonal velocity, 500 hpa meridional velocity, 500 hpa velocity, 500 hpa geopotential height, 500 hpa wind direction, 500 hpa divergence, 850 hpa airflow strength, 850 hpa zonal velocity, 850 hpa meridional velocity, 850 hpa velocity, 850 hpa geopotential height, 850 hpa wind direction, 850 hpa divergence, Relative humidity at 500 hpa, Relative humidity at 850 hpa, Near surface relative humidity, Surface specific humidity and Mean temperature at 2m. All 26 variables were selected as the predictor variables [4].

The dataset was normalized over the complete period with respect to their 1961-1990 means and standard deviations. The first 25 years (1961-1985) was selected as training data, next eight years (1986-1993) was taken as validation data and the final eight years (1994-2001) was taken as test data.

3.2 Creating the ANN Pool

ANN pool was created using a set of GRNN trained with different training data. Different training data was obtained by preprocessing the input data set and using moving block bootstrap [19].

Reddy, Neralla and Gidson has identified a relationship between the sunspot cycle and the Indian ocean rainfall [20]. Their results show that rainfall has 11 year cycles. Using this information another time series was created to forecast a day's rainfall using 11 years previous data. Rainfall also has one year cycles from which another time series was created. The original data series and the two created data series were sampled using moving block bootstrap with block length (b) 365, chosen to represent the number of days in a year. The details are summarized in Table 1.

Table 1. Moving Block Bootstrap

Data Series	No. of Records in the Series (n)	No. of Possible Adjacent Blocks (n-b+1)	No. of Blocks Sampled (n/b)	No. of Possible Bootstrap Time Series	No. of GRNN Trained
Original	9125	8761	25	350	350
11 Year	5110	4746	14	339	328
1 Year	8760	8396	24	349	345

The data blocks for the bootstrap time series were selected without replacement and using those time series a total of 1023 GRNN were trained which made the ANN pool. The trained GRNN were assigned a unique number from 1 - 1023 so that they can be identified. ANN for all models described in the next section were selected from this pool.

3.3 Selecting the Members and Combining the Ensembles

To increase the generalization ability of the ensemble most accurate and diverse members have to be selected while maintaining the balance between them.

Two techniques are proposed to achieve this ENN-K and ENN-GA. To compare their performance bagging and boosting models are developed. All models are described in detail in this section.

ENN-K Model. Diversity can be described as the difference in ANN error when the same input data is given. To identify the most diverse ANN the ANN were clustered according to their RMSE on validation data using k-means clustering algorithm. The errors of ANN in the same cluster are similar and the errors of ANN in different clusters are different. As a result ANN in each cluster are diverse from the ANN in other clusters. The algorithm for creating ENN-K model is as follows.

```
Cluster the ANN in the pool according to their RMSE for the validation set using k-means clustering algorithm.
Select a suitable value for number of clusters k (eg. 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 20, 30, 40, 50, 75, 100).
Initialize k means vectors at random, $\mu_i$, (i=1, 2, …, k).
Classify the input vectors according to the closest means vectors $\mu_i$, to k clusters.
Re-compute $\mu_i$.
If there are any changes in each $\mu_i$, for all input vectors, classify the input vectors again. Otherwise stop.
Select the ANN with the smallest RMSE from each cluster.
Combine the selected ANN with a GRNN and train and test the ensemble.
Repeat all the steps for different values for k.
Select the ensemble with the smallest RMSE as the final ensemble.
```

ENN-GA Model. In this model the ANN for the ensemble were selected using a binary GA to minimize the RMSE of the ensemble. The algorithm for creating ENN-GA model is given described here. Figure 1 describes the basic genetic algorithm.

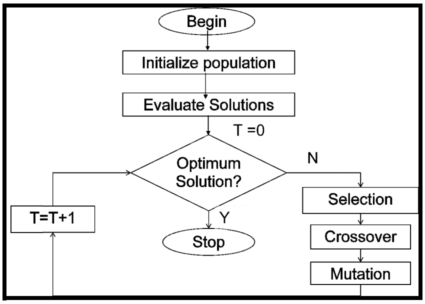


Fig. 1. Basic genetic algorithm

The GA begins by defining GA parameters as described in Table 2.

Table 2. Genetic Algorithm Parameters

Parameter	Value		
Chromosome length	100	150	200
Population size	102	68	51
Maximum number of ANN per chromosome	10	15	20
Mutation probability	0.1		
Crossover probability	0.6		
Selection	Roulette Wheel Rank Weighting		

The initial population was created by a set of chromosomes. A chromosome represents the list of ANN in the ensemble by their numbers. 10 bits were used to represent one ANN. Figure 2 represents part of an ANN in binary and after decoding.

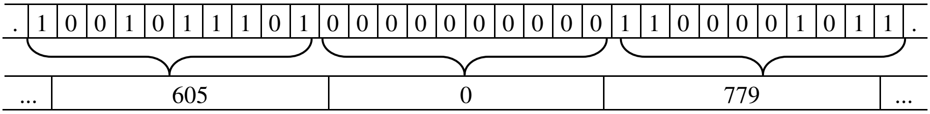


Fig. 2. Part of a chromosome in binary and its decoded version

To decode the chromosome Each 10 bit binary number was converted in to decimal to find the corresponding ANN. A zero means no network. A decoded chromosome of length 150 is depicted in Figure 3. It represents an ensemble with 12 ANN.

150	203	44	12	0	2	51	718	657	1011	0	70	810	0	540

Fig. 3. A decoded chromosome of length 150

For each chromosome in the population an ensemble with the given ANN was created using a GRNN as the combiner and RMSE of the ensemble was found. The chromosomes were ranked and sorted according to their cost. Then, the mating pool was prepared by selecting only the best. Two chromosomes were selected from the mating pool to produce two new offspring. Pairing takes place in the mating population until enough offspring were born to replace the discarded chromosomes. Mating took place using one point crossover. The offspring were added to the mating pool to make the new population. For mutation 10% of the bits in the new population were inverted randomly. The procedure from finding cost for each chromosome was repeated until the ensemble with minimum RMSE is found. The algorithm was repeated for three different sizes of chromosomes as described in Table 1.

3.4 Validating the Models

To validate the ENN-K and ENN-GA models and compare their performances ensembles were created using two widely used methods bagging and boosting. Their algorithms are given below.

Bagging. A number of ANN were taken from the pool with replacement and an ensemble was created with average method.

Boosting. A number of ANN were selected from the pool without replacement and an ensemble was created with weighted average method. Another ensemble was created by taking some other ANN from the pool half of which were ANN in previous ensemble that had the lowest weights. This process was repeat. The main difference between bagging and boosting is that in boosting the observations are weighted to encourage better predictions for points that were previously misclassified.

Also to check the models performance with other datasets all the developed models were tested with rainfall data for Katugastota, located in the western coast of Sri Lanka on North latitude 7.33° and East longitude 80.62º. Average annual rainfall of Katugastota is about 1800 mm.

4 Results and Discussion

All the models in our study were developed using daily weather data for 41 years from 1961 to 2001 with 25 years training data 8 years validation data and 8 years testing data. All the results shown in this section are from testing the models with testing data. Our results shows that the proposed two methods outperform the bagging and boosting models for both Colombo and Katugastota.

ENN-K model was developed by testing with different cluster sizes and the bagging and boosting models were developed by testing with different numbers of ANN in the ensemble as follows 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 20, 30, 40, 50, 75, 100. Figure 4 shows the RMSE obtained for ENN-K, bagging and boosting models for different numbers of ANN in the ensemble for both Colombo and Katugastota. The best results were obtained by ENN-K with 12 and 15 GRNN in the ensemble, bagging with 17 and 16 GRNN in the ensemble and boosting with 11 and 13 GRNN in the ensemble for Colombo and Katugastota.

ENN-GA model was developed by implementing the algorithms for three different sizes of chromosomes separately. The best result was obtained by ENN-GA model with 14 GRNN in the ensemble for Colombo and 12 GRNN in the ensemble for Katugastota with chromosome size 150. Table 3 shows the performance of ENN-GA for the three chromosome lengths.

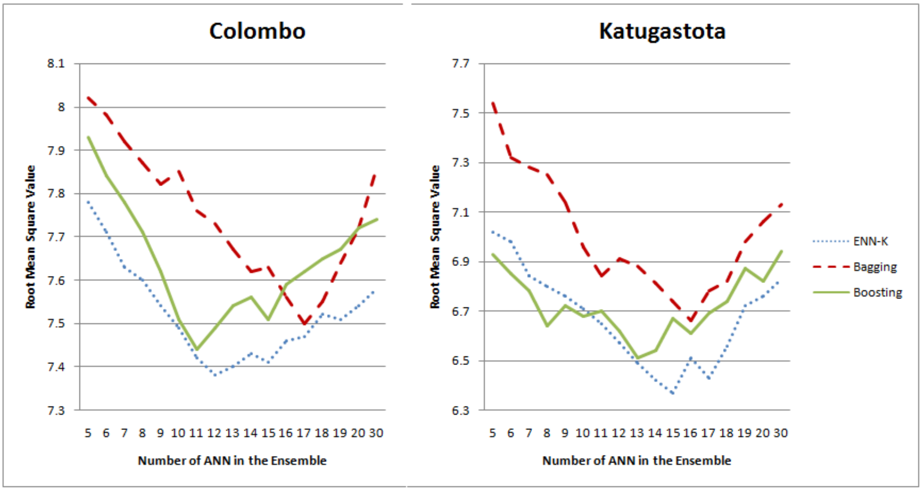


Fig. 4. Performance of ENN-K, bagging and boosting models for different sizes of ensembles

Table 3. Genetic Algorithm Performance for different chromosome lengths

Parameter	Chromosome Length					
	Colombo			Katugastota		
	100	150	200	100	150	200
Number of ANN in best performing ensemble	10	14	14	10	12	12
RMSE of the best performing ensemble	7.49	7.33	7.36	6.33	6.25	6.31
Number of generations for convergence	392	285	338	286	238	254

A comparison of the performance of each model by the Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and the Coefficient of Determination (R^2) for both cities are shown in Table 4.

Table 4. Performance of the Best Ensemble in Each Model

Model	Colombo				Katugastota			
	No. of ANN in ensemble	RMSE	MAE	R^2	No. of ANN in ensemble	RMSE	MAE	R^2
ENN-K	12	7.38	4.32	0.673	15	6.37	4.18	0.613
ENN-GA	14	7.33	4.24	0.676	12	6.25	4.02	0.633
Bagging	17	7.50	4.42	0.652	16	6.66	4.41	0.564
Boosting	11	7.44	4.37	0.657	13	6.51	4.22	0.599

In creating ENN-K model only one GRNN, with the smallest RMSE was selected from a single cluster. But it may be that there are more than one suitable member in a given cluster.

The bagging and boosting methods are two widely used methods that have been validated through numerous research as good ensemble creation methods. The ENN-K and ENN-GA models gave more accurate results than the bagging and boosting models with smaller RMSE and MAE values and larger R^2 values for both Colombo and Katugastota. This implies that the proposed two models can be used to select more suitable members for ensembles. In this study the models are developed for rainfall forecasting and tested with a single dataset.

5 Conclusion

This study introduce two methods ENN-K and ENN-GA to select the most appropriate members for a neural network ensemble developed for rainfall forecasting. Colombo, Sri Lanka was chosen as the study area for this study and the implemented ANN models were trained, validated and tested using daily observed weather data of 41 years. The performance of the developed models were compared with the performance of two widely used methods for ensemble creation bagging and boosting.

Our results indicate that the two proposed methods can outperform bagging and boosting models in selecting members for ensembles for rainfall forecasting. Out of the two proposed models ENN-GA performs better than ENN-K.

References

1. ASCE Task Committee on the Application of Artificial Neural Networks in Hydrology.: Artificial neural networks in hydrology: II. Hydrologic applications. Journal of Hydrologic Engineering (2), 124–137 (2000)
2. Hansen, L.K., Salamon, P.: Neural network ensembles. IEEE Trans. Pattern Anal. 12(10), 993–1001 (1990)
3. Nagahamulla, H.R.K., Ratnayake, U.R., Ratnaweera, A.: An ensemble of Artificial Neural Networks in Rainfall Forecasting. In: International Conference on Advances in ICT for Emerging Regions, pp. 176–181. IEEE Press, Colombo (2012)
4. Nagahamulla, H.R.K., Ratnayake, U.R., Ratnaweera, A.: Artificial Neural Network Ensembles in Time Series Forecasting: an Application of Rainfall Forecasting in Sri Lanka. International Journal on Advances in ICT for Emerging Regions 6(2) (2013)
5. Nagahamulla, H.R.K., Ratnayake, U.R., Ratnaweera, D.A.A.C.: An Effective Neural Network Ensemble Architecture For Short Term Rainfall Forecasting. In: Special Session on Urban Water Environment Monitoring & Management 4th International Conference on Structural Engineering and Construction Management, Kandy, pp. 58–68 (2013)
6. Zhou, Z., Wu, J., Tang, W.: Ensembling Neural Networks: Many Could Be Better Than All. Artificial Intelligence 137(1-2), 239–263 (2002)
7. Sharkey, A.J.C., Sharkey, N.E.: Combining Diverse Neural Nets. Knowledge Engineering Review 12(3), 299–314 (1997)
8. Gheyas, I.A., Smith, L.S.: A Neural Network Approach to Time Series Forecasting. In: Proceedings of the World Congress on Engineering, vol. II (2009)
9. Maqsood, I., Khan, M.R., Abraham, A.: An Ensemble of Neural Networks for Weather Forecasting. Neural Computing and Applications 13, 112–122 (2004)

10. Sharkey, A.J.C.: Combining Artificial Neural Nets - Ensemble and Modular Multi Net Systems. Springer (1999)
11. Mitchell, M.: An Introduction to Genetic Algorithms. MIT Press, Cambridge (1996)
12. Opitz, D., Shavlik, J.W.: Actively Searching for an Effective Neural Network Ensemble. Connection Science 8 (1996)
13. Kima, H., Kimb, H., Moonc, H., Ahnb, H.: A Weight-Adjusted Voting Algorithm for Ensemble of Classiers. Journal of the Korean Statistical Society (2011)
14. Wolpert, D.: Stacked generalization. Neural Networks 5, 241–259 (1992)
15. Sylvester, J., Chawla, N.V.: Evolutionary Ensembles: Combining Learning Agents using Genetic Algorithms. American Association for Artificial Intelligence (2005)
16. Nazir, M., Jaffar, M.A., Hussain, A., Mizra, M.: Efficient Gender Classification using Optimization of Hybrid Classifiers using Genetic Algorithm. International Journal of Innovative Computing, Information and Control 7(12) (2011)
17. Bauter, E., Kohavi, R.: An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. Machine Learning 36, 105–142 (1999)
18. Kalnay, E., et al.: The NCEP/NCAR 40-year reanalysis project. Bulletin of the American Meteorological Society 77, 437–471 (1996)
19. Efron, B., Tibshirani, R.J.: An Introduction to the Bootstrap. Chapman & Hall/CRC (1998)
20. Reddy, R.S., Neralla, V.R., Gidson, W.L.: The Solar Cycle and Indian Rainfall. Theoretical and Applied Climatology 39(4), 194–198

Simulating Basic Cell Processes with an Artificial Chemistry System

Chien-Le Goh[1], Hong Tat Ewe[2], and Yong Kheng Goh[2]

[1] Faculty of Computing and Informatics, Multimedia University, Malaysia
clgoh@mmu.edu.my
[2] Faculty of Engineering and Science, Universiti Tunku Abdul Rahman, Malaysia
{eweht,gohyk}@utar.edu.my

Abstract. When we simulate life, there are always more things to simulate than what have been coded. Life is complex and seems to have endless possibilities. Using artificial chemistry as a starting point to simulate life is a promising way to limit the possibilities because chemical reactions are always the same under the same physical conditions. We have built an 3D artificial chemistry system simulating molecules and the reactions among them. Our goal is to simulate a cell or a group of cells in the future using mainly molecules and chemical reactions. In this paper, we show that the system can simulate the fundamental aspects of reproduction, metabolism and adaptation in cells. This is accomplished by simulating reproducing molecules, reactions which provide energy to the reproducing molecules and the adaptation ability of reproducing molecules.

Keywords: artificial chemistry, cell process simulation.

1 Introduction

There are three branches of natural computing [1]. They are computing inspired by nature, synthesis of natural phenomena and computing with natural materials. Artificial life is an area of studies under the second branch. Its main goal is to study life by creating artificial entities which have life-like behaviours. Artificial life researchers have used many different approaches in their researches. There are, for example, researches to simulate and study the energy system, the evolution, the healing process, the growth process and the movement of life forms. The targetted life forms can come from a wide range, from abstract creatures, birds, insects, organs to cells.

The use of artificial chemistry [2] to simulate life is a promising approach. When we simulate life, there are always more things to simulate than what have been coded. Life is complex and seems to have endless possibilities. Using artificial chemistry as a starting point to simulate life is a promising way to limit the possibilities because chemical reactions are always the same under the same physical conditions.

We have built an 3D artificial chemistry system [3] for simulating molecules and the reactions among them. Our goal is to simulate a biological cell or a group

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_57, © Springer International Publishing Switzerland 2014

of cells in the future using molecules and chemical reactions for the purpose of studying the computation models of cells.

In this paper, we show that the system can simulate the fundamental aspects of reproduction, metabolism and adaptation. This is accomplished by simulating reproducing molecules, reactions which provide energy to the reproducing molecules and the adaptation ability of reproducing molecules.

The next section describes some notable related work followed by the artificial chemistry system employed, the experimental results and the conclusion.

2 Related Work

In [4], the author proposed a cell simulation model based on artificial chemistry. The model has a simple scheme of representing seven types of atoms, labelled a to f. Each atom can be in a state numbered 0, 1, 2, ... A state number can be changed through an artificial chemical reaction. The atoms move randomly in the simulated environment. A reaction involves one or two reactants and can be described in the forms such as $a3 + b4 \rightarrow a6b6$ and $c3d7 \rightarrow c6 + d6$. This model has been implemented as a Java applet called the Organic Builder and has been made accessible through a web site for the public to experiment [5]. The model has shown that artificial chemistry can be an effective tool for cell simulation.

A model which is able to generate complete organisms possessing metabolism and morphology from a single initial cell was proposed in [6] and later extended in [7] to use L-systems [1]. It is a hybrid model combining artificial chemistry and L-systems. The model simulates a cell growing into an organism of cells in a 2-D grid which contains chemical molecules. Chemical reactions in this model consume or produce energy beside producing new chemical molecules.

In [8], an artificial chemical reaction optimization algorithm (ACROA) was proposed. The algorithm uses chemical reactants and nature inspired chemical reactions to operate on the population of reactants to incrementally find the optimal solution to a problem. In the algorithm, a population of reactants (chemical molecules) are initially encoded to represent a set of random solutions. Operations which mimic chemical reactions are then applied to the reactants to produce new solutions. The possible reactions are synthesis reactions, decomposition reactions, single displacement reactions, double displacement reactions, combustion reactions, redox reactions and reversible reactions. The population of reactants are evaluated after applying the reactions. The reactions and the evaluations are applied repeatedly until a termination criterion is fulfilled. ACROA was used by the authors to solve a multiple sequence alignment problem and to mine association rules to demonstrate its possible uses.

A model of chemical reactions optimization (CRO) was proposed as a method to solve optimization problems in [9]. In the model, a population of possible solutions are represented as a population of molecules with different molecular structures. A molecule in CRO possesses two kinds of energy, potential energy and kinetic energy. The former is a quantity derived from the molecular structure while the latter a measure of tolerance for the molecule changing to a different structure. It represents the ability of a molecular structure to escape from a local minimum. The initial population is allowed to interact to search for structures with lower and lower levels of potential energy. Eventually, the structure with the lowest potential energy when the stop condition is met is presented as the solution for the problem. Four types of elementary reactions are allowed in the model. They are on-wall ineffective collision, decomposition, inter-molecular ineffective collision and synthesis. The reactions cause the molecules to change to different structures. The loss of potential energy is converted to kinetic energy stored in a central energy buffer. The stored energy is in turn used when needed to facilitate decomposition and synthesis. The model was used to solve the quadratic assignment problem, a resource-constrained project scheduling problem and a channel assignment problem.

The work described above has shown that chemical reactions can be simulated in an abstract manner using artificial chemistry and artificial chemistry has the potential to model cells. In addition it has shown that artificial chemistry also has characteristics which can be used to compute a solution to a problem. Our artificial chemistry system is an extension of the work done in [4]. We have extended the simulation space from 2-D to 3-D, added support for molecules and taken into consideration the energy aspect of artificial chemistry.

3 The Artificial Chemistry System

The 3D artificial chemistry system used to conduct experiments in this paper consists of two parts, a simulator and a visualizer. The molecule model, the reaction model the simluated environment of the system is explained in [3]. Briefly described here, The simulated environment is an open environment where molecules can enter the simulation space from outside the boundaries and leave the simulation space if they move beyond the boundaries. Energy is utilized or released during chemical reactions. For the ease of simulation, a variable is used to keep track of global energy which is assumed to be instantly distributed evenly throughout the entire simulation space.

The simulator reads three input files: the molecule declaration file, the reaction definition file and the molecule in-flow file, and generates a Mathematica notebook for each time step and a log keeping track of the number of molecules and the global energy in the system. Mathematica is used as the visualizer to view the simulation space which contains molecules (see Fig. 1).

4 Experiments

Reproduction, metabolism and adaptation are three of the basic processes in a cell. In this section, we use artificial chemistry to simulate reactions which can reproduce molecules, reactions which can provide energy to other reactions and reactions which enable molecules to adapt to changes in the types of molecules injected into the system. This aim is to show that the fundamental aspects of the three processes can be simulated.

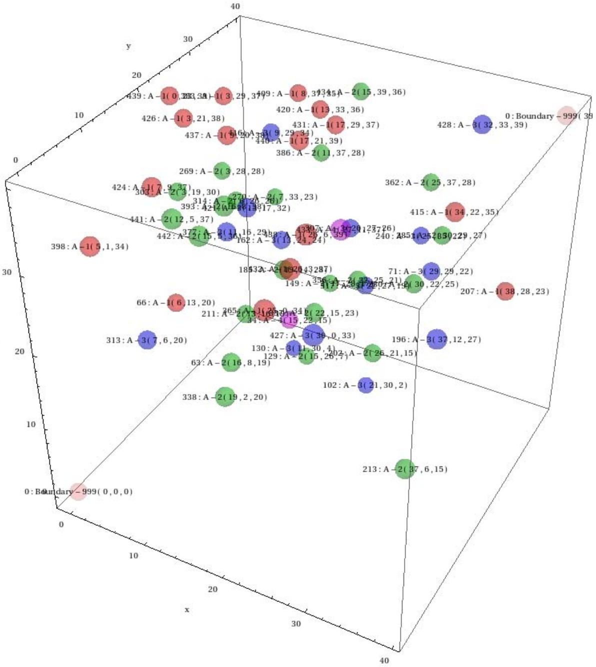


Fig. 1. The simulation space viewed with Mathematica

All the experiments below were run within a simulated 3D space of 64,000 unit^3 (40 x 40 x 40). Global energy level was set to 64,100 units so that there was an excess of 100 unit of energy to kick-start reactions.

4.1 Reactions to Reproduce Molecules

We used a set of simple reactions to simulate reproducing molecules as shown in Table 1. A-2 molecules were chosen arbitratily as the molecules to reproduce. The first reaction produces an A-2 molecules from two A-1 molecules. The second reaction produces an A-3 molecule from an A-2 molecule and an A-1 molecule and so on. The fourth reaction produces two A-2 molecules from an A-4 molecule. The focus here is to keep producing A-2 molecules from A-4 molecules. This set of reactions is an abstraction of the reactions involved in the reproduction of cells. A cell consumes nutrients to become a matured cell which in turn produces more cells.

Table 1. Reactions to reproduce A-2

ID = 1	ID = 2	ID = 3	ID = 4
Energy = 0	Energy = 0	Energy = 0	Energy = 0
A-1 + A-1 ->	A-2 + A-1 ->	A-3 + A-1 ->	A-4 ->
A-2 .	A-3 .	A-4 .	A-2 + A-2 .

In order to test the viability of the reactions, we measured the number of A-2 molecules in the system. Fig. 2 shows the number of A-1, A-2, A-3 and A-4 molecules when the simulator was run for 1000 steps. The molecule in-flow consisted of only A-1 molecules. At every time step, five A-1 molecules were injected into the system from the top of the simulated space.

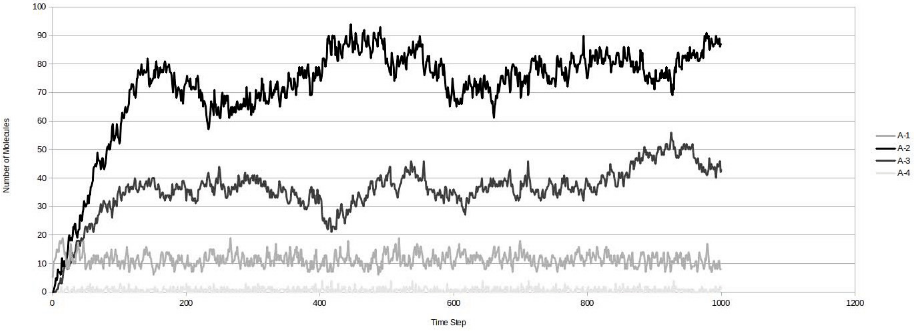


Fig. 2. Number of A-1, A-2, A-3 and A-4 molecules over time

Relative to other molecules, A-2 molecules were the most abundant and the number of A-2 molecules stabilized over time. This shows that the set of reactions used was able to produce a stable reproductive cycle. The number of injected A-1 moleculdes and the number steps in a reproductive cycle can affect the reproductive cycle. We conducted more experiments to study the effects and the results are shown in Fig. 3 and Fig. 4.

In Fig. 3, when the number of injected molecules increased, the number of A-2 molecules also increased and in each case the number of A-2 molecules stabilized over time. The stability of the reproductive cycle was not affected by the number of injected molecules. We then fixed the number of injected molecules (A-1) to ten and increased the number of steps in a reproductive cycle to produce A-3 molecules and A-4 molecules with the reactions listed in Table 2 and Table 3. As shown in Fig. 4, the reproductive cycle stabilized even with the increase in the number of steps. From the results, we conclude that the set of reproductive reaction rules used here can be used in the simulation of cells in the future to produce a stable cell population.

4.2 Energy Providing Reactions

All the reactions used in the experiments in the previous sub-section neither use nor produce energy. From the results obtained in the previous sub-section, the most successful set of reactions, the set which reproduced A-2 molecules, was chosen to investigate the amount of energy needed to reproduce in a stable manner. Two new reactions were added into Table 1 to produce a new set of reactions in Table 4.

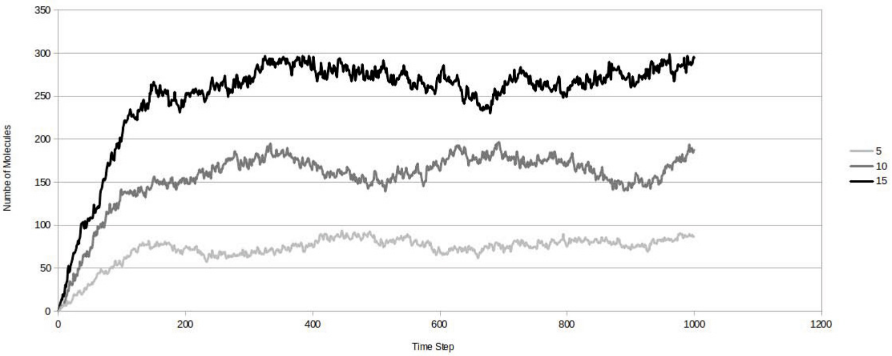


Fig. 3. Number of A-2 molecules when five, ten and fifteen A-1 molecules are injected every time step

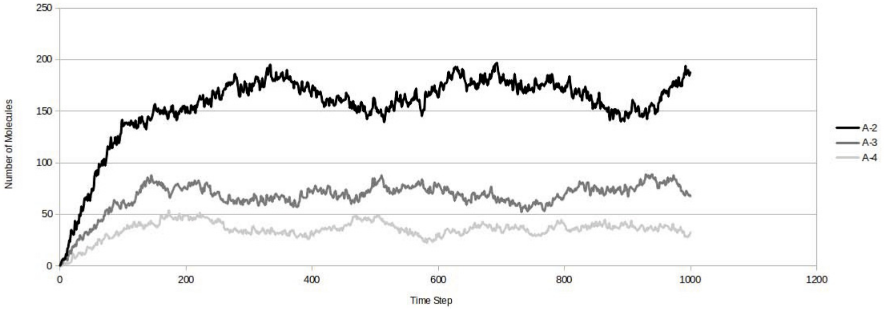


Fig. 4. Number of A-2, A-3 and A-4 molecules with different sets of reactions (ten A-1 molecules are injected every time step for each case)

Table 2. Reactions to reproduce A-3

ID = 1
Energy = 0
A-1 + A-1 ->
A-2 .
	ID = 2
	Energy = 0
	A-2 + A-1 ->
	A-3.
		ID = 3
		Energy = 0
		A-3 + A-1 ->
		A-4.
			ID = 4
			Energy = 0
			A-4 + A-1 ->
			A-5 .

ID = 5
Energy = 0
A-5 + A-1 ->
A-6 .
	ID = 6
	Energy = 0
	A-6 ->
	A-3 + A-3 .		

Table 3. Reactions to reproduce A-4

ID = 1
Energy = 0
A-1 + A-1 ->
A-2 .
	ID = 2
	Energy = 0
	A-2 + A-1 ->
	A-3.
		ID = 3
		Energy = 0
		A-3 + A-1 ->
		A-4.
			ID = 4
			Energy = 0
			A-4 + A-1 ->
			A-5 .

ID = 5
Energy = 0
A-5 + A-1 ->
A-6 .
	ID = 6
	Energy = 0
	A-6 + A-1 ->
	A-7 .
		ID = 7
		Energy = 0
		A-7 + A-1 ->
		A-8 .
			ID = 8
			Energy = 0
			A-8 ->
			A-4 + A-4 .

Table 4. The set of reactions to reproduce A-2 with energy producing reactions

ID = 1
Energy = -1
A-1 + A-1 ->
A-2 .
	ID = 2
	Energy = -1
	A-2 + A-1 ->
	A-3 .
		ID = 3
		Energy = -1
		A-3 + A-1 ->
		A-4 .
			ID = 4
			Energy = -1
			A-4 ->
			A-2 + A-2 .

ID = 5
Energy = -1
A-1 + OK-1 ->
A-1 + K-1 .
	ID = 6
	Energy = 4
	K-1 + OK-1 ->
	OP-1 .		

Reactions 5 and 6 in Table 4 mimics metabolism. When an A-1 molecule senses the presence of an OK-1 molecule in its vicinity, it produces a K-1 molecule. The K-1 molecule can react with an OK-1 molecule to release energy which can be used to sustain reproduction. OP-1 molecules are the waste products. Reactions 1 to 5 each was defined to use one unit of energy. We varied the amount of energy released by reaction 6, from 0 to 7 units, and conducted simulations to find the right amount of energy to set to create a stable system. The results are shown in Fig. 5. It shows the number of A-2 molecules for each variation of reaction 6. The molecule in-flow consisted of only A-1 molecules and OK-1 molecules. The probabilities of the insertion of OK-1 molecules and the insertion of A-1 molecules were both set to 0.5. The number of molecules inserted into the system every time step was set to ten.

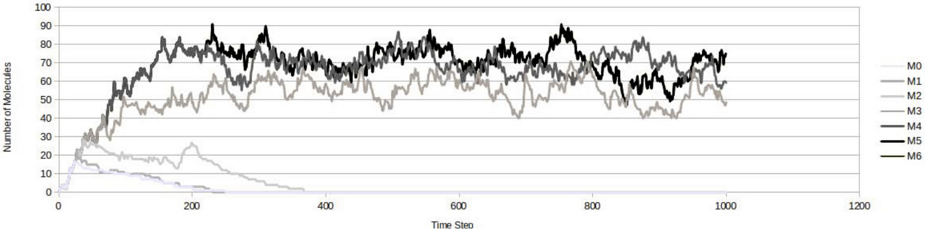


Fig. 5. The number of A2 molecules produced when the energy released by reaction 6 is varied from 0 to 6

When the energy released by reaction 6 was 0, 1, 2, or 3 units the reproduction process eventually stopped due to insufficient energy in the system. When the energy released by reaction 6 was 4 units or more, the reproduction process continued on throughout the duration of a simulation. Although the higher the energy released by reaction 6, the more A-2 molecules were produced until all the energy needs of reactions 1 to 5 were fulfilled, we need to find the right balance where the local energy stabilizes to close to 1 unit. In other words, we do not want a system which always has increasing local energy, nor a system which always has decreasing local energy.

Therefore, we also measured local energy, which is the amount of energy distributed evenly in each unit space, when the energy released by reaction 6 varied

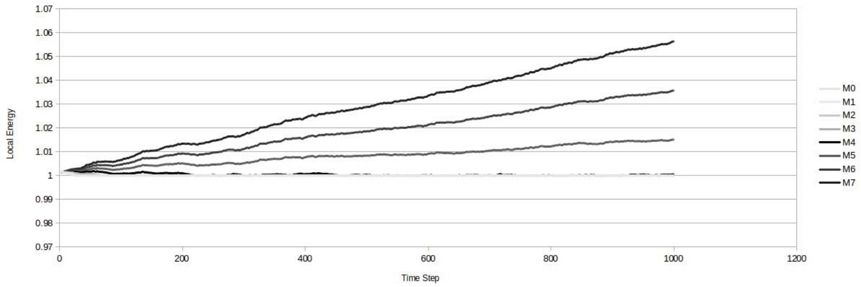


Fig. 6. Local energy when the energy released by reaction 6 is varied from 0 to 7

from 0 to 7 (see Fig. 6). Local energy kept increasing when the energy released was 5 and above. The best result was achieved when the energy released was 4. At that level, the reproduction process of `A-2` molecules had sufficient energy while local energy stabilized at around 1.

4.3 Reactions for Adaptation

To experiment with adaptation, we created three different groups of reactions and put them together as listed in Table 5. Reactions 1 to 4 are about the reproduction of `A-2` molecules, reactions 5 to 6 are about releasing energy from `OK-1` molecules and reaction 7 is about `A-1` molecules changing to `B-1` molecules to adapt to the new in-flow of `OL-1` molecules replacing `OK-1`. Reactions 8 to 14 are similar to reactions 1 to 7 but they cater to the reproduction of `B-2` molecules the adaptation to new in-flow of `OM-1` molecules replacing `OM-1` molecules. Reactions 15 to 20 are about the reproduction of `C-2` molecules.

Experiment were conducted with the in-flow of `A-1` and only one type of either `OK-1`, `OL-1` or `OM-1` molecules at any one time. Similar to the previous subsection, the probability of the insertion of each molecule type was set to 0.5. The number of molecules inserted into the system every time step was set to ten. However, at time step 1000, `OK-1` molecules were replaced by `OL-1` molecules and subsequently at time step 1500, `OL-1` molecules were replaced by `OM-1` molecules. The goal is to show that the reaction set in Table 5 can adapt to the changes in the molecule type inserted into the system and keep the reproduction process going.

After experimenting with differennt combinations of energy values, we found that when the energy values of reactions 6, 13 and 20 were set to 5, 9 and 9, we could obtain the results shown in Fig. 7. The fluctuation of local energy was very low increasing only from 1 to 1.02 (see Fig. 8). `A-2` molecules, `B-2` molecules and `C-2` molecules were produced in direct response to the changes in the molecule type inserted.

Table 5. The set of reactions for reproduction with reactions to adapt

ID = 1
Energy = -1
A-1 + A-1 ->
A-2 .
	ID = 2
	Energy = -1
	A-2 + A-1 ->
	A-3 .
		ID = 3
		Energy = -1
		A-3 + A-1 ->
		A-4 .
			ID = 4
			Energy = -1
			A-4 ->
			A-2 + A-2 .

ID = 5
Energy = -1
A-1 + OK-1 ->
A-1 + K-1 .
	ID = 6
	Energy = 5
	K-1 + OK-1 ->
	OP-1 .
		ID = 7
		Energy = -1
		A-1 + OL-1 ->
		B-1 .	

ID = 8
Energy = -1
B-1 + B-1 ->
B-2 .
	ID = 9
	Energy = -1
	B-2 + B-1 ->
	B-3 .
		ID = 10
		Energy = -1
		B-3 + B-1 ->
		B-4 .
			ID = 11
			Energy = -1
			B-4 ->
			B-2 + B-2 .

ID = 12
Energy = -1
B-1 + OL-1 ->
B-1 + L-1 .
	ID = 13
	Energy = 9
	L-1 + OL-1 ->
	OQ-1 .
		ID = 14
		Energy = -1
		A-1 + OM-1 ->
		C-1 .	

ID = 15
Energy = -1
C-1 + C-1 ->
C-2 .
	ID = 16
	Energy = -1
	C-2 + C-1 ->
	C-3 .
		ID = 17
		Energy = -1
		C-3 + C-1 ->
		C-4 .
			ID = 18
			Energy = -1
			C-4 ->
			C-2 + C-2 .

ID = 19
Energy = -1
C-1 + OM-1 ->
C-1 + M-1 .
	ID = 20
	Energy = 9
	M-1 + OM-1 ->
	OR-1 .		

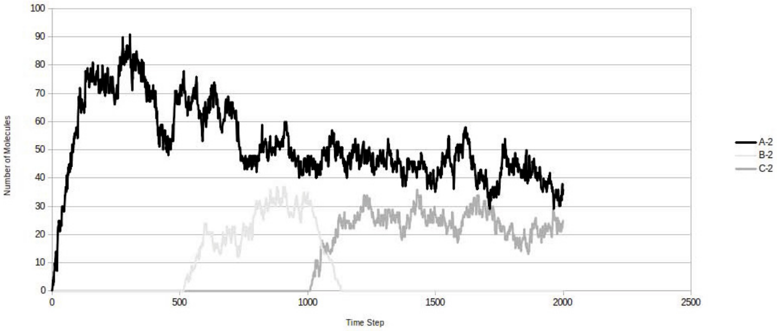


Fig. 7. The adaptation process with the energy units of reactions 6, 13 and 20 set to 5, 9 and 9

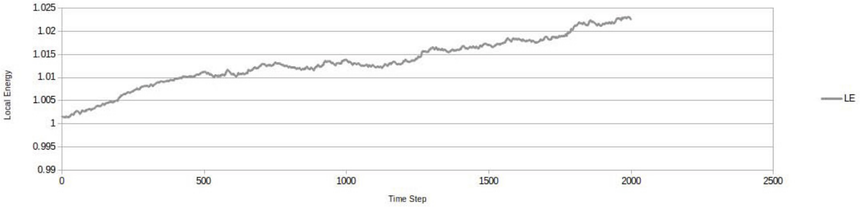


Fig. 8. Local energy with the energy units of reactions 6, 13 and 20 set to 5, 9 and 9

5 Conclusion

With the use of chemical reactions only in our artificial chemistry model, we have simulated reproducing molecules, reactions which provide energy to the reproducing molecules and the adaptation ability of reproducing molecules. Although the results are still far from complete simulations of reproduction, metabolism and adaptation in cells, we have shown the potential in using chemical reactions and molecules to simulate cell processes.

A systematic comparison between the actual cell processes and the reactions listed in this paper is still needed. To simulate a complete cell, other cell processes such as genetic encoding and decoding, the immune system, the self-repair mechanism and their bio-chemical reactions sill need to be explored.

References

1. Castro, L.: Fundamentals of Natural Computing: An Overview. Physics of Life Reviews 4, 1–36 (2007)
2. Dittrich, P., Ziegler, J., Banzhaf, W.: Artificial Chemistries - A Review. Artificial Life 7(3), 225–275 (2001)
3. Goh, C., Ewe, H.T., Goh, Y.K.: An Artificial Chemistry System for Simulating Cell Chemistry: The First Step. In: Tan, Y., Shi, Y., Mo, H. (eds.) ICSI 2013, Part I. LNCS, vol. 7928, pp. 32–39. Springer, Heidelberg (2013)
4. Hutton, T.J.: Evolvable Self-reproducing Cells in a Two-dimensional Artificial Chemistry. Artificial Life 13(1), 11–30 (2007)
5. Hutton, T.J.: The Organic Builder A Public Experiment in Artificial Chemistries and Self-Replication. Artificial Life 15(1), 21–28 (2009)
6. Cussat-Blanc, S., et al.: From Single Cell to Simple Creature Morphology and Metabolism. In: Bullock, S., Noble, J., Watson, R., Bedau, M. (eds.) Artificial Life XI, pp. 134–141 (2008)
7. Djezzar, N., et al.: L-systems and Artificial Chemistry to Develop Digital Organisms. In: IEEE Symposium on Artificial Life, pp. 225–232 (2011)
8. Alatas, B.: ACROA: Artificial Chemical Reaction Optimization Algorithm for Global Optimization. Expert Systems with Applications 38(10), 13170–13180 (2011)
9. Lam, A., Li, V.: Chemical-Reaction-Inspired Metaheuristic for Optimization. IEEE Transactions on Evolutionary Computation 14(3), 318–399 (2010)

The Effectiveness of Sampling Methods for the Imbalanced Network Intrusion Detection Data Set

Kok-Chin Khor[1], Choo-Yee Ting[1], and Somnuk Phon-Amnuaisuk[2]

[1] Faculty of Computing and Informatics, Multimedia University, Jalan Multimedia,
63100, Cyberjaya, Selangor, Malaysia
{kckhor,cyting}@mmu.edu.my
[2] Faculty of Business and Computing, Brunei Institute of Technology,
Mukim Gadong A, BE1410, Brunei Darussalam
somnuk.phonamnuaisuk@itb.edu.bn

Abstract. One of the countermeasures taken by security experts against network attacks is by implementing Intrusion Detection Systems (IDS) in computer networks. Researchers often utilize the *de facto* network intrusion detection data set, KDD Cup 1999, to evaluate proposed IDS in the context of data mining. However, the imbalanced class distribution of the data set leads to a rare class problem. The problem causes low detection (classification) rates for the rare classes, particularly R2L and U2R. Two commonly used sampling methods to mitigate the rare class problem were evaluated in this research, namely, (1) under-sampling and (2) over-sampling. However, these two methods were less effective in mitigating the problem. The reasons of such performance are presented in this paper.

Keywords: Imbalanced KDD Cup 1999 Data set, Sampling, Rare Class Problem.

1 Introduction

Collecting data from busy computer networks for IDS usually results in a huge data set with a highly imbalanced class distribution. An attack class such as Denial-of-Service (DoS) could overwhelm other classes in a data set as it usually comes in a huge amount. The *de facto* data set for network intrusion detection domain, KDDCup 1999 data set, shows such imbalanced characteristic [1]. Such characteristic causes the rare class problem, where classifying rare classes become difficult.

In a real-life problem, the ratio of rare classes to dominant classes can be 1 to 100, 1 to 1,000, 1 to 10,000 or even more [2]. Generally, the problem of classifying rare classes in an imbalanced data set could be mitigated through (1) under-sampling and (2) over-sampling methods [3]. The under-sampling method reduces the size of dominant classes in a data set, whereas the over-sampling method increases the size of rare classes in a data set. Both methods are non-heuristic with the aim to equalize the class distribution of a data set. However, there are drawbacks employing these two methods [4]. The under-sampling method may remove useful records in dominant classes. Nevertheless, the computational cost can be reduced especially if the dominant

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_58, © Springer International Publishing Switzerland 2014

classes are huge in size. The over-sampling method, conversely, could increase computational cost and subsequently introduce over-fitting problem because of the duplicate copies of rare classes' records in a data set.

In this research, the effectiveness of these two sampling methods in mitigating the rare class problem was evaluated using the *de facto* network intrusion data set. In Section 2, an overview on the data set shall be discussed. Section 3 provides the empirical results obtained using these two sampling methods and a result comparison with a similar research. The results of applying these sampling methods shall be explained in Section 4. Section 5 concludes the research and suggests possible mitigations for improving the detection rates for the rare classes.

2 Data Set Overview

Although the *de facto* data set has received critics from researchers [5, 6], it is widely used by data mining researchers in the IDS domain [7-14]. There are reasons why the data set remains popular until today. Firstly, there is a lack of better public IDS test set [15]. Organizations are reluctant to reveal their computer networks to others worrying that it may expose its network vulnerabilities to intruders. Secondly, generalizing a huge raw network data for data mining purposes requires a Herculean effort from experts. In addition, the tool MADAM ID, which is used to construct the features of the *de facto* data set, is no longer available [26].

The *de facto* data set is a result of DARPA intrusion detection evaluation program [1]. A Local Area Network (LAN) environment was simulated based on U.S Air Force LAN and it was hit continuously with various attacks for nine weeks. The raw TCP dump data was collected from the LAN and transformed into a useful data set. It was then used in 1999 Knowledge Discovery and Data Mining Competition. All attacks in the data set can be categorized into one of these four attack classes, namely, Denial-of-service (DoS), Probing (Probe), Remote-to-local (R2L), and User-to-root (U2R). The normal network traffic in the data set is labeled as Normal class.

Table 1. The distribution of the classes in the 10% KDDCup 1999 data set

Class	**Training set (*trs*)**	**Testing set (*tes*)**
Normal	97,277	60,593
DoS	391,458	229,853
Probe	4,107	4,166
R2L	1,126	16,189
U2R	52	228
Total	**494,020**	**311,029**

The *de facto* data set has a training set and a testing set with size approximately to 5 million and 2 million records, respectively. The size of each record is about 100 bytes and both sets give a total of 700 Megabytes. The huge data set requires a high processing power for data mining purposes. Much IDS research, as well as this research, preferred a reduced version of the *de facto* data set (10% of the original data set) provided by the competition organizer [8, 9, 12, 16, 17]. The data set has 41 features, comprising both intrinsic and derived features. The intrinsic features are essential features that can be obtained from the TCP dump data. The derived features, on the other hand, are formed based on experts' opinion.

Table 1 shows the class distribution of the reduced data set. The training set (denoted as *trs*) shows a heavily imbalanced class distribution, where 19.7% and 79.2% of the records belong to Normal and DoS classes, respectively. It leaves only 1.1% to the other three classes (Probe, R2L and U2R). Since rare classes have fewer records, the dominant classes might overwhelm them. Any classification models built using this data set may not be effective in detecting the rare classes but may give good detection rates for the dominant classes.

The testing set (denoted as *tes*) does not follow the probability distribution of the training set and additional attacks are added to the classes to make IDS evaluation more realistic. An attack named *snmpgetattack*, which takes 47.7% of the R2L class in the testing set, is undistinguishable from the Normal class [18]. Further, this attack cannot be found in the training set. In short, this research would expect low detection rates when attempting to detect R2L using the testing set.

3 Under-Sampling and Over-Sampling

An empirical study was conducted to investigate whether under-sampling and over-sampling have any effect on the detection rates for the rare classes. The study was conducted using WEKA [19].

Table 2. Comparison of the original training set (*trs*) with the under-sampled training set (*trs_us*) of KDD Cup 1999

Class	Training Set (*trs*)	Under-sampled Training Set (*trs_us*)
Normal	97,277	87,830
DoS	391,458	54,572
Probe	4,107	4,107
R2L	1,126	1,126
U2R	52	52
Total	**494,020**	**147,687**

The dominant classes, Normal and DoS were under-sampled to balance the class distribution of the *trs* by removing identical records of the dominant classes and resulted in another training set, *trs_us*. As shown in Table 2, a reduction size of 86.1% and 9.7% was achieved for Normal and DoS classes, respectively. Because of the reduction, the size of the *trs* was greatly reduced and only 147,687 records (29.9% of *trs*) remained in *trs_us*.

The empirical study also investigated the effectiveness of Synthetic Minority Over-sampling Technique (SMOTE), an over-sampling technique proposed by [20] on *trs_us*. SMOTE is created for imbalanced data sets, where it involves the creation of synthetic rare examples along the line joining the randomly selected *k* rare neighbors. The default *k* value, five, was used in this research. The number of R2L and U2R were increased from 50% to 1000% for *trs_us*. The data set was discretized and the dimension of *trs* and *trs_us* were reduced using the optimal feature set determined in our previous work [21]. The data set dimension was reduced to save computational cost during the learning phase.

The data set *trs* was evaluated using major learning algorithms such as Naïve Bayes Classifier (NBC), Bayesian Networks (BN), and Decision Trees algorithms such as ID3, J48 (or C4.5) and Classification and Regression Trees (CART). Robust leaning algorithms such as Support Vector Machine (SVM) and Artificial Neural

Table 3. Detection rates (%) achieved using major learning algorithms on *trs*. The numbers in bold indicate the very low detection rates on the rare classes.

Normal	***DoS***	***Probe***	***R2L***	***U2R***	**Algorithm**
99.3	96.3	86.0	**12.0**	**19.7**	NBC
99.5	94.0	83.5	**5.5**	**17.1**	BN
99.3	96.2	82.5	**1.1**	**11.4**	ID3
98.2	96.8	74.1	**3.6**	**13.2**	J48
99.5	93.5	89.4	**7.0**	**15.8**	CART

Table 4. Detection rates (%) achieved using major learning algorithms on *trs_us*. The numbers in bold indicate the very low detection rates on the rare classes, R2L and U2R.

Normal	**DoS**	**Probe**	**R2L**	**U2R**	**Algorithm**
99.4	93.7	79.1	**11.2**	**19.3**	NBC
98.2	94.1	82.5	**8.0**	**16.7**	BN
99.3	99.7	87.8	**1.5**	**29.3**	ID3
99.5	96.8	77.7	**0.7**	**13.2**	J48
99.3	97.5	78.1	**3.6**	**14.5**	CART

Table 5. Detection rates (%) achieved using ID3, where R2L and U2R classes of *trs_us* were over-sampled using SMOTE. The numbers in bold indicate the very low detection rates on the rare classes, R2L and U2R.

Normal	***DoS***	***Probe***	***R2L***	***U2R***	**Remarks**
99.3	99.7	87.8	**1.5**	**29.3**	Rare classes +50%
99.3	99.7	87.8	**1.5**	**29.3**	Rare classes +100%
99.3	99.7	87.8	**1.5**	**29.3**	Rare classes +150%
99.3	96.3	83.1	**15.7**	**10.7**	Rare classes +200%
99.7	96.3	83.1	**15.5**	**11.7**	Rare classes +400%
99.3	96.3	83.1	**15.5**	**10.6**	Rare classes +600%
99.3	96.3	83.1	**15.5**	**10.6**	Rare classes +800%
99.3	96.3	83.1	**15.5**	**10.6**	Rare classes +1000%

Network (ANN) were not included as they were computational infeasible for the data set, even though the data set size had been reduced significantly. Overall, the learning algorithms did not perform well in detecting R2L and U2R classes as expected (Table 3). But satisfactory detection rates were achieved for another rare class, Probe. High detection rates were also achieved for Normal and DoS classes.

The empirical study was then continued using the under-sampled data set, *trs_us*. Although the dominant classes Normal and DoS were greatly reduced, the detection rates for R2L and U2R were still very low (Table 4). Attempt was then made to improve the detection rates by over-sampling these two rare classes in *trs_us* using SMOTE (over-sampled from 50% to 1000% to their original sizes). ID3 was used in this attempt as it gave the highest overall detection rates among the learning algorithms on *trs* (not shown in Table 4). As shown in Table 5, the detection rate for R2L increased after over-sampling these two rare classes for 200%. Nevertheless, the detection rate for U2R class was dropped from 29.3% to 10.7%. The detection rates for these two rare classes stagnated after over-sampling 600% to their original sizes (R2L – 15.5% and U2R – 10.6%). The results suggested an over-fitting problem in *trs_us* after increasing the size of rare classes to a very huge extend.

3.1 Compare with a Similar Research

A study by [22] used under-sampling and over-sampling methods as well in their research. Initially, a wrapper approach was implemented to find the optimal percentage of under-sampling to the training set for every considerate amount of dominant classes. Then, a search of optimal over-sampling percentage came with fixed percentage of under-sampling. As shown in Table 6, their results conformed to the empirical results of this research, where poor detection rates were attained for R2L and U2R classes.

Table 6. The detection result of a similar research [22]. The numbers in bold indicate the very low detection rates on the rare classes, R2L and U2R.

Class	Detection Rate (%)
Normal	95.6
DoS	97.3
Probe	91.9
R2L	**13.7**
U2R	**19.7**

4 Reasons of the Unsatisfactory Detection Results

There are two reasons why both sampling methods are less effective in detecting R2L and U2R classes. Firstly, the *de facto* data set shows a characteristic that prohibited the learning algorithms from recognizing R2L and U2R classes well. Secondly, many major learning algorithms have weaknesses dealing with imbalanced data sets.

4.1 Overlapping Classes

The *trs* were examined using matrix plots by showing the pairwise relationship of any two features and also the decision boundary for each class in the data set. A decision boundary is a volume in a feature space which is determined by a classification algorithm [23]. All vectors in a decision region are considered belong to the same class. A total of 1681 matrix plots were generated using 41 features of the data set. Only two were selected for discussion. All matrix plots show the similar phenomenon as discussed below.

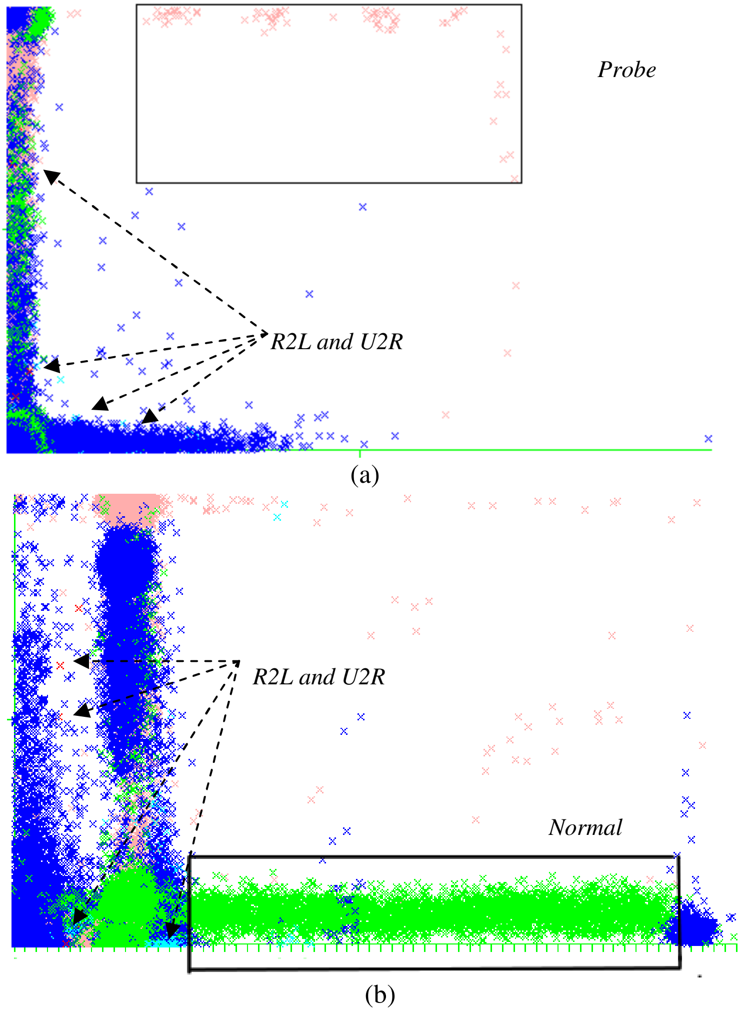


Fig. 1. Two matrix plots are shown where (a) involves features “duration” and “dst_host_rerror_rate” and (b) involves features “service” and “dst_host_diff_srv_rate”. The five classes in the data set are represented using crosses with different colors. As shown by both matrix plots, both *R2L* (light blue crosses) and *U2R* classes (red crosses) are overwhelmed by others. The squares in (a) and (b) show possible decision boundaries for *Probe* class (purple crosses) and *Normal* class (green crosses), respectively.

As shown in Fig. 1(a) and Fig. 1(b), R2L (light blue crosses) and U2R (red crosses) were overlapped with other classes, especially Normal (green crosses) and DoS (blue crosses). Because of the overlapping, the decision boundaries of these two rare classes could not be generalized clearly by a learning algorithm. Low detection rates were, therefore, attained for them. Another rare class, Probe could be detected relatively easy as compared to R2L and U2R because its decision boundary could be generalized easily (refer to the square in Fig.1 (a)). Therefore, satisfactory detection results were obtained for Probe. High detection rates were expected for Normal and DoS classes using any major learning algorithm because of their dominance in the data set.

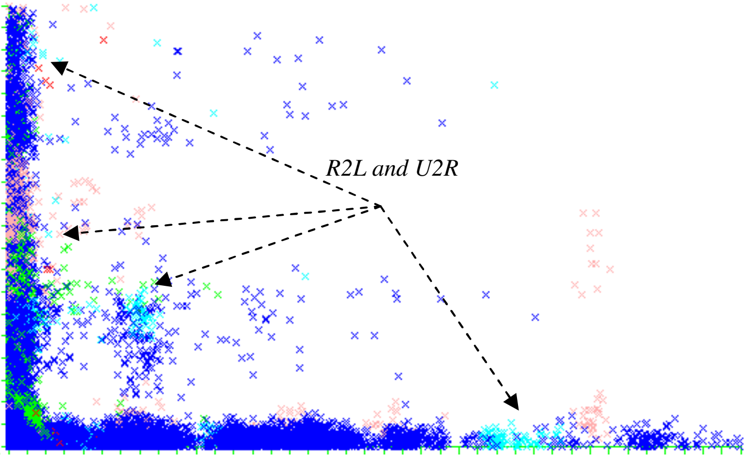


Fig. 2. The matrix plot shows the pairwise relationship between the features "duration" and "dst_host_rerror_rate" of *trs_us*. The dominant classes, especially *Normal* (blue crosses) and *DoS* (green crosses), still outnumbered *R2L* (light blue crosses) and *U2R* (red crosses) classes after under-sampling process.

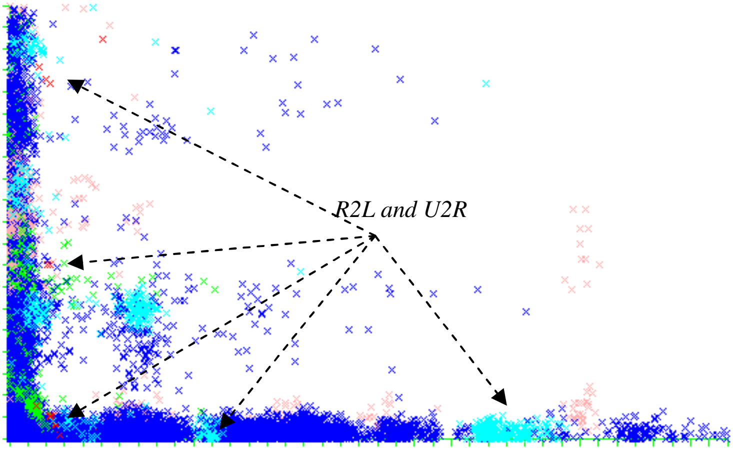


Fig. 3. The pairwise relationship between the features "duration" and "dst_host_rerror_rate" of *trs_us* after increasing the original size of the rare classes to 200% using SMOTE. *R2L* (light blue crosses) is less overwhelmed by the dominant classes as compared to *U2R* (red crosses).

The unsatisfactory detection results using under-sampling and over-sampling methods are explained using matrix plots, generated using *trs_us*. Fig. 2 shows that even though the dominant classes had reduced significantly after under-sampling, they still overlapped with R2L (light blue crosses) and U2R classes (red crosses). Therefore, the decision boundaries for these two rare classes were difficult to generalize. On the other hand, there was less improvement even though these two rare classes were increased significantly in size using SMOTE. Fig. 3 shows that these two rare classes, especially U2R (red crosses) were still overwhelmed by the others. In short, the overlapping of classes still occurred regardless of the sampling methods used on the data set.

4.2 Weaknesses of Learning Algorithms

The phenomenon discussed in Sections 4.1 demonstrated the challenges of existing learning algorithms dealing with the imbalanced *de facto* data set; in this context, the R2L and U2R classes. Many learning algorithms aim to minimize the overall error, where rare classes give minimum contribution [24]. Therefore, learning algorithms are biased to the dominant classes and against the rare classes. The learning algorithms also work on assumption that the positive and negative examples are roughly the same and the error costs of different classes are assumed to be the same. Other factors such as improper evaluation metrics used, data fragmentation problem, use of inappropriate inductive bias for data generalization, and noisy data also cause difficulties in mining imbalanced data set [2, 25].

5 Conclusions

This research investigated the effectiveness of two common sampling methods namely, under-sampling and over-sampling for the rare class problem inherited in the *de facto* data set. These two methods are less effective because of the overlapping classes in the *de facto* data set. In addition, the dominant classes far outnumber the rare classes. Major learning algorithms were unable to generalize decision boundaries for R2L and U2R as learning algorithms generally favor the dominant classes.

The overlapping classes could also create difficulties in the feature selection phase. Any feature subset identified may not be able to give good detection rates. In our previous work [21], several feature selection methods, including *filter* and *wrapper* approaches were attempted to find the optimal feature set for the data set. The result was satisfactory in overall but low detection rates were attained for *R2L* and *U2R* classes.

Suggestions to mitigate the rare class problem are as follows based on the characteristics of the imbalanced data set.

1. To consider *algorithm-level approach* that makes learning algorithms suitable for the imbalanced data set without manipulating the class distribution as sampling methods [3]. The approach introduces bias based on the imbalanced data set and trains a classification model for each rare class (classifier specific).

2. To consider a *divide-and-conquer* approach, which is to separate the rare classes from the dominant classes and form cascaded classifiers to detect network attacks [9]. This approach reduces the problem of overlapping classes and helps generalizing a clear decision boundary for the rare classes.
3. To collect more data for *R2L* and *U2R* classes instead of creating synthetic data using techniques like SMOTE.

References

1. ACM KDD Cup 1999. Computer Network Intrusion Detection (1999), `http://www.sigkdd.org/kddcup/`
2. Chawla, N.V., Japckowicz, N., Kolcz, A.: Editorial: Special Issue on Learning from Imbalanced Data sets. ACM SIGKDD Explorations Newsletter 6(1), 1–6 (2004)
3. Kotsiantis, S., Kanellopoulos, D., Pintelas, P.: Handling Imbalanced Data sets: A Review. GESTS International Transactions on Computer Science and Engineering 30, 25–36 (2006)
4. Chawla, N.V.: Data Mining for Imbalanced Data sets: An Overview. In: Data Mining and Knowledge Discovery Handbook, 2nd edn., pp. 875–886. Springer Science + Business Media (2000)
5. McHugh, J.: Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory. ACM Transactions on Information and System Security 3(4), 262–294 (2000)
6. Brugger, S.T., Chow, J.: An assessment of the DARPA IDS Evaluation Data set using Snort. Technical Report CSE-2007-1, University of California, Department of Computer Science, Davis, CA (2007)
7. Horng, S.J., Su, M.Y., Chen, Y.H., Kao, T.W., Chen, R.J., Lai, J.L., Perkasa, C.D.: A Novel Intrusion Detection System Based on Hierachical Clustering and Support Vector Machine. Expert Systems with Applications 38, 306–313 (2011)
8. Gupta, K.K., Nath, B.: Layered Approach Using Conditional Random Fields for Intrusion Detection. IEEE Transaction on Dependable and Secure Computing 7(1), 35–49 (2010)
9. Khor, K.C., Ting, C.Y., Phon-Amnuaisuk, S.: A Cascaded Classifier Approach for Improving Detection Rates on Rare Attack Categories in Network Intrusion Detection. Applied Intelligence 36, 320–329 (2012)
10. Li, Y., Wang, J.L., Tian, Z.H., Lu, T.B., Young, C.: Building Lightweight Intrusion Detection System Using Wrapper-based Feature Selection Mechanisms. Computers & Security 28(6), 466–475 (2009)
11. Depren, O., Topallar, M., Anarim, E., Kemal Ciliz, M.: An Intelligent Intrusion Detection System (IDS) for Anomaly and Misuse Detection in Computer Networks. Expert Systems with Applications 29(4), 713–722 (2005)
12. Xiang, C., Png, C.Y., Lim, S.M.: Design of Multiple-level Hybrid Classifiers for Intrusion Detection System Using Bayesian Clustering and Decision Trees. Pattern Recognition 29(7), 918–924 (2008)
13. Liu, G., Yi, Z., Yang, S.: A Hierarchical Intrusion Detection Model Based on the PCA Neural Networks. Neurocomputing 70(7-9), 1561–1568 (2007)
14. Agarwal, R., Joshi, M.V.: PNRule: A New Framework for Learning Classifier Models in Data Mining (A Case-Study in Network Intrusion Detection). Technical Report, No. RC-21719, IBM Research Division (2001)

15. Engen, V., Vincent, J., Phalp, K.: Exploring Discrepancies in Findings Obtained with the KDD Cup '99 Data Set. Journal of Intelligent Data Analysis 15(2), 251–276 (2011)
16. Hu, W.M., Hu, W., Maybank, S.: Adaboost-Based Algorithm for Network Intrusion Detection. IEEE Transaction on Systems, Man, and Cybernetics-Part B 38, 577–583 (2008)
17. Pfahringer, B.: Winning the KDD99 Classification Cup: Bagged Boosting. SIGKDD Explorations 1, 65–66 (2000)
18. Bouzida, Y., Cuppens, F.: Detecting Known and Novel Network Intrusions. In: Fischer-Hübner, S., Rannenberg, K., Yngström, L., Lindskog, S. (eds.) Security and Privacy in Dynamic Environments. IFIP, vol. 201, pp. 258–270. Springer, Boston (2006)
19. The University of Waikato, Weka 3, `http://www.cs.waikato.ac.nz/ml/weka/`
20. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research 16, 321–357 (2002)
21. Khor, K.C., Ting, C.Y., Phon-Amnuaisuk, S.: Forming an Optimal Feature Set for Classifying Network Intrusions Involving Multiple Feature Selection Methods. In: International Conference on Information Retrieval and Knowledge Management, pp. 178–182 (2010)
22. Chawla, N.V., Hall, L.O., Joshi, A.: Wrapper-based computation and evaluation of sampling methods for imbalanced data sets. In: Proceedings of the 1st International Workshop on Utility-Based Data Mining, pp. 24–33 (2005)
23. de Sá, J.P.M.: Pattern Recognition: Concepts, Methods And Applications. Springer, New York (2001)
24. Visa, S., Ralescu, A.: Issues in Mining Imbalanced Data Sets - A Review Paper. In: Sixteen Midwest Artificial Intelligence and Cognitive Science Conference, pp. 67–73 (2005)
25. Weiss, G.M.: Mining with Rarity: A Unifying Framework. ACM SIGKDD Explorations Newsletter 6(1), 7–19 (2004)
26. Bouzida, Y.: Principal Component Analysis for Intrusion Detection and Supervised Learning for New Attack Detection. PhD Thesis (2006)

A Clustering Based Technique for Large Scale Prioritization during Requirements Elicitation

Philip Achimugu, Ali Selamat, and Roliana Ibrahim

UTM-IRDA Digital Media Centre, K-Economy Research Alliance &
Faculty of Computing, Universiti Teknologi Malaysia,
Johor Baharu, 81310, Johor, Malaysia
check4philo@gmail.com, {aselamat,roliana}@utm.my

Abstract. We consider the prioritization problem in cases where the number of requirements to prioritize is large using a clustering technique. Clustering is a method used to find classes of data elements with respect to their attributes. K-Means, one of the most popular clustering algorithms, was adopted in this research. To utilize k-means algorithm for solving requirements prioritization problems, weights of attributes of requirement sets from relevant project stakeholders are required as input parameters. This paper showed that, the output of running k-means algorithm on requirement sets varies depending on the weights provided by relevant stakeholders. The proposed approach was validated using a requirement dataset known as RALIC. The results suggested that, a synthetic method with scrambled centroids is effective for prioritizing requirements using k-means clustering.

Keywords: Software, requirements, weights, prioritization, clustering.

1 Introduction

During software development process, there are more prospective requirements specified for implementation with limited time and resources. Therefore, a meticulously selected set of requirements must be considered for implementation with respect to available resources [1]. The process of selecting preferential requirements for implementation is referred to as requirements prioritization. This process aims at determining an ordered relation on specified sets of requirements [2].

There are so many advantages of prioritizing requirements before implementation. First, prioritization aids the implementation of a software system with preferential requirements of stakeholders [3]. Also, the challenges associated with software development such as limited resources, inadequate budget, insufficient skilled programmers among others makes requirements prioritization really important. It can help in planning software releases since not all the elicited requirements can be implemented in a single release due to the challenges associated with software development [4]. Consequently, determining which, among pool of requirements to be implemented first and the order of implementation is a critical success factor in software development.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_59, © Springer International Publishing Switzerland 2014

To prioritize requirements, stakeholders will have to compare them in order to determine their relative value through preference weights [5]. These comparisons grow with increase in the number of requirements [6]. State-of-the-art prioritization techniques such as AHP and CBRanks seem to demonstrate high capabilities [7]. These techniques have performed well in terms of ease of use and accuracy but, still lacking in scalability and rank reversals respectively. Rank reversals refer to the ability to update or reflect rank status anytime an attribute is added or deleted from a set. In this paper, an enhanced approach for software requirements prioritization is proposed based on the limitations of existing approaches.

2 Related Work

Different prioritization techniques have been proposed in the literature. According to the research documented in [8], existing prioritization techniques are classified under two main categories, which include: techniques that are applied to small number of requirements (small-scale) and techniques that applied to larger number of requirements (medium-scale or large-scale). Examples of small-scale techniques include round-the-group prioritization, multi-voting system, pair-wise analysis, weighted criteria analysis, and the quality function deployment approach. However, techniques for prioritizing larger number of requirements include: MoSCoW, binary priority list, planning game, case based rank and the wiegers's matrix approaches.

A further classification of existing prioritization techniques was provided by [9]. They similarly divided existing techniques into two main categories: (1) techniques which enable values or weights to be assigned by project stakeholders against each requirement to determine their relative importance and (2) methods that include negotiation approaches in which requirements priorities result from an agreement among subjective evaluation by different stakeholders. Examples of techniques that apply to the first category are analytical hierarchy process (AHP), cumulative voting, numerical assignment, planning game and wieger's method. An example of the second category would be the win-win approach and the multi criteria preference analysis requirement negotiation (MPARN).

The most cherished and reliable prioritization technique as reported in the literature is the AHP technique; although it also suffers scalability problems with increase in the number of requirements. An in-depth analysis and descriptions of existing prioritization techniques with their limitations can be found in [10]. Nonetheless, obvious limitations that cut across existing techniques ranges from rank reversals to scalability, inaccurate rank results, increased computational complexities and unavailability of efficient support tools among others. However, this research seeks to address most of these limitations with the aid of clustering algorithms.

3 The Proposed Approach

Clustering is an optimization problem where the aim is to partition a given set of data objects into a certain number of clusters in order to determine the relative closeness between those objects [11]. In this paper, we concentrated on the development of a large-scale prioritization approach using k-means, where the numbers of requirement sets (*R*), constructed clusters (*k*), and attributes (*A*) are relatively huge. K-means utilizes a two-phased iterative algorithm to reduce the sum of point-to-centroid distances, summed over all *k* clusters described as follows: The first phase implores the "batch" updates to re-assign points to their nearest cluster centroid, which initiates the re-calculation of cluster centroids. The second phase uses the "online" updates to re-assign points so as to reduce the sum of distances which causes the re-computation of cluster centroids after each reassignment. In this research, the former was adopted because; the clusters are updated based on minimum distance rule. That is, for each entity *i* in the data table, its distances to all centroids are calculated and the entity is assigned to the nearest centroid. This process continues until all the clusters remain unchanged. Before loading the datasets for the algorithm to run, there is need to pre-process or standardized them.

K-Means is an unsupervised clustering method that is applicable to a dataset represented by set of *N* to *Ith* entity with set of *M* to *Vth* feature. Therefore, the entity-to-feature matrix *Y* will be given by (*yiv*), where *yiv* is the feature value $v \in V$ at entity $i \in I$. This process generate a partition $S = \{S_1, S_2, \ldots, S_K\}$ of *I* in *K* non-overlapping classes *Sk*, referred to as clusters. Each of these cluster have specific centroids denoted as *ck* = (*ckv*) with an *M*-dimensional vector in the feature space (k=1, 2,…K). Centroids form set $C = \{c_1, c_2, \ldots, c_K\}$.The criterion, minimized by this method, is the within-cluster summary distance to the centroids. A partition clustering can be characterized by (1) the number of clusters, (2) the cluster centroids, and (3) the cluster contents. Thus, we used criteria based on comparing either of these characteristics in the generated data with those in the resulting clustering while; the centroids are calculated by finding the average of the entries within clusters.

During requirements prioritization, the project stakeholders converge to assign weights to requirements. Before weights assignment takes place, the elicited requirements are described to the relevant stakeholders in order to understand each requirement and the implication of weighting one requirement over the other. Therefore, the main aim of this research is to propose a technique of prioritizing requirements based on the preference weights provided by the stakeholders. The metric distance function was utilized in approximating the distances between each requirement weight. These requirements can thus be considered as points in a *K* dimensional Euclidean space. The aim of the clustering in this research work is to minimize the intra-cluster diversity (distortion) when ranking or prioritizing large requirements.

The case presented in this paper has to do with the calculation of relative importance of requirement sets across relevant stakeholders based on the preferential weights of attributes contained in each set. These weights are partitioned into clusters with the help of centroids to determine the final clusters of requirement sets based on the Euclidean space of each attribute weight. The cluster centroids are responsible for attracting requirements to their respective clusters based on a defined criterion. Prioritization can therefore be achieved by finding the average weights across attributes in all the clusters. For instance, if we have requirement sets as $R = \{r_1 \ , r_2, \ldots, r_k | i = 1, \ldots, N\}$ of dimensional attributes *A*, defined by $(a_1, a_2, \ldots, a_K)$ over 5 stakeholders. Prioritization will mean computing all the relative weights of attributes provided by stakeholders based on a weighting scale over each requirement set. These requirement sets are partitioned into various clusters given as $K = \{k_1, k_2, \ldots, k_M\}$. Each cluster will contain the relative weights of all the stakeholders for a particular requirement set. The algorithm is described below:

```
1. Initialize $m_i$, $i = 1,\ldots, n$, for example, to k random $x^t$
2. Repeat
3. For all $x^t$ in X
        i. $k_i^t \leftarrow 1$ if $\| x^t - m_i \| = \min_j \| x^t - m_j \|$
        ii. $k_i^t \leftarrow 0$ otherwise
4. For all $m_i$, $i = 1,\ldots,n$
        i. $m_i \leftarrow$ sum over t $(k_i^t x^t)$ / sum over t $(k_i^t)$
5. Until $m_i$ converge
```

Algorithm 1. Computation of relative weights

The vector *m* contains attribute weights with mean under each cluster, while *X* stands for the centroids and *k* represent the estimated cluster labels. The algorithm executes as follows:

1. It will select a pattern in which to initialize m_i to form clusters, and do it.
2. For each attribute in a requirement set, the algorithm captures the weights provided by the stakeholders for that set and assigns it to a new cluster (represented by m_i).
3. For each m_i, a new centroid is calculated by finding the average of the weights and the cluster is re-calculated to reflect the mean relative weights of the set.
4. Steps 2-3 are repeated until m_i converges.

Therefore, each cluster k_i (i=1,…, n) has requirements classified by the centroid. Rank reversals can be addressed by calculating a new centroid and mean each time an attribute is added or deleted from the list. The mean of a given requirement set is:

$$\overline{Z}^i = \left(\sum\nolimits_{r \in k_i} R\right) \Big/ |n_i| \tag{1}$$

Assuming, the requirements are points of a Euclidean space, the normalization σ of weights in a cluster is defined as:

$$\sigma = \frac{1}{n_i} \sum_{i=1}^{N} \overline{Z}^i \tag{2}$$

However, Equation (3) is used to compute the distance or disagreement measures between requirement sets. This is achieved by computing the mean distance of attributes in each requirement set with respect to their cluster centroids.

$$d = \sum_{k=1}^{K} \left(a_k^{(i)} - a_k^{(j)}\right) \tag{3}$$

Equation 4, which is the square root of the variance, is used to prioritize requirements.

$$P = \sqrt{\sum_{k=1}^{K} \left(a_k^{(i)} - a_k^{(j)}\right)} \tag{4}$$

We ran the straight K-Means algorithm for different weights, *W* of attributes *A* in a range from START value (typically 1, in our experiments) to END value (typically,

10) with respect to the number of stakeholders *S* (Algorithm 2). The average weights of each cluster is obtained and normalized. Given a cluster *K*, the smallest *W(S, A)* is subtracted from the largest and the square root of the difference is obtained to reflect the overall relative weights of each requirement set (Equations 3 and 4).

```
K-Means Results Generation
For K=The number of clusters START: END
    For diff_init=1: number of different K-means initializations
        randomly select K entities as initial centroids and normalize
        run Straight K-Means algorithm
        calculate the WK, the value of W(S, A)
        for each K , take the average W among different clusters
        compute the disagreement values and find its square root
    end diff_init
end K
```

Algorithm 2. Requirement prioritization process

A solution to a clustering problem can be depicted by a partitioning table and cluster centroids. These two techniques are intertwined; that is, if one is given, the optimal choice of the second one can be uniquely generated. However, this is executed based on two optimal conditions:

a. Nearest neighbour condition: The attributes for a given set of cluster centroids can be optimally classified by assigning it to a cluster with the closest centroid.
b. Centroid condition: The disagreement of the optimal cluster representative given in a partition is minimized with the help of the centroid of the cluster members.

Clustering problems can be addressed by using either the centroid based (CB) technique or partition based (PB) technique. However, in this research, the CB technique was adopted. In centroid-based technique, the centroid *X* for a given set of requirement is determined by summing all the attributes in the cluster, divided by their numbers. Each cluster is visited at least once to avoid erroneous results. The weights in each cluster is computed in a greedy way is to ensure efficient processing of clusters with large numbers of attributes and to minimize inter-cluster discrepancies. Each cluster in the solutions is assigned a number and cluster size, indicating the number of attributes that belongs to it.

The proposed technique for requirements prioritization was enhanced by applying a few steps of the k-means algorithm for each new solution. This operation first generates a rough estimate of the solution which is then refined by the k-means algorithm. This modification allows faster convergence of the solution.

4 Experimental Setup

The experiments described in this research investigated the possibility of computing preference weights of requirements across all stakeholders in a real-world software project using k-means algorithm. As mentioned previously, the metrics evaluated in this experiment are (1) the number of generated clusters, (2) the cluster centroids, and

(3) the cluster contents. The RALIC datasets was used for validating the proposed approach. The PointP, RateP and RankP aspect of the requirement datasets were used, which consist of about 262 weighted attributes spread across 10 requirement sets from 76 stakeholders. RALIC stands for replacement access, library and ID card [12]. It was a large-scale software project initiated to replace the existing access control system at University College London. The datasets are available at: http://www.cs.ucl.ac.uk/staff/S.Lim/phd/dataset.html. Attributes were ranked based on 5-point scale; ranging from 5 (highest) to 1 (lowest). As a way of pre-processing the datasets, attributes with missing weights were given a rating of zero [13].

For the experiment, a Gaussian Generator was developed, which computes the mean and standard deviation of given requirement sets. It uses the Box-Muller transform to generate relative values of each cluster based on the inputted stakeholder's weights. The experiment was initiated by specifying a minimum and maximum number of clusters, and a minimum and maximum size for attributes. It then generates a random number of attributes with random mean and variance between the inputted parameters. Finally, it combines all the attributes into one and computes the overall score of attributes across the number of clusters *k*. The algorithms defined earlier attempt to use these combined weights of attributes in each cluster to rank each requirement set. For the k-means algorithm to run, we filled in the variables/observations table which has to do with the three aspect of RALIC dataset that was utilized (PointP, RateP and RankP), followed by the specification of clustering criterion (Determinant W) as well as the number of classes. The initial partition was randomly executed and ran 50 times. The iteration completed 500 cycles and the convergence rate was at 0.00001.

5 Experimental Results

The results displayed in Table 1 shows the summary statistics of 50 experimental runs. In 10 requirement sets, the total number of attributes was 262 and the size of each cluster varied from 1 to 50 while, the mean and standard deviation of each cluster spanned from 1-30 and 15-30, respectively.

Table 1. Summary statistics

Variables	Obs.	Obs. with missing data	Obs. with missing data	Min	Max	Mean	Std. deviation
Rate P	262	0	262	0.000	262	5.123	15.864
Point P	262	0	262	2.083	262	28.793	24.676
Rank P	262	0	262	0.000	262	1.289	16.047

*Obs. = Objects

Also, Figure 1 shows the results of running the clustering algorithm on the data set when trying to find 10 clusters. It displays the generated 10 clusters which represent 10 sets of requirements with various numbers of weighted attributes and the within-class variance. Figure 2 shows the statistics summary of the experimental iteration. The error function value was within 3.5.

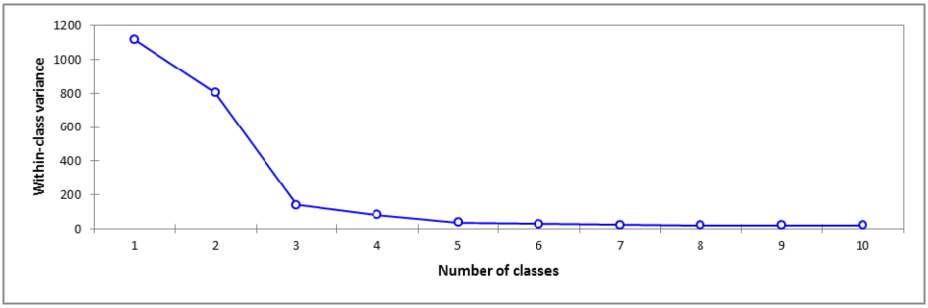


Fig. 1. Evolution of variances within classes

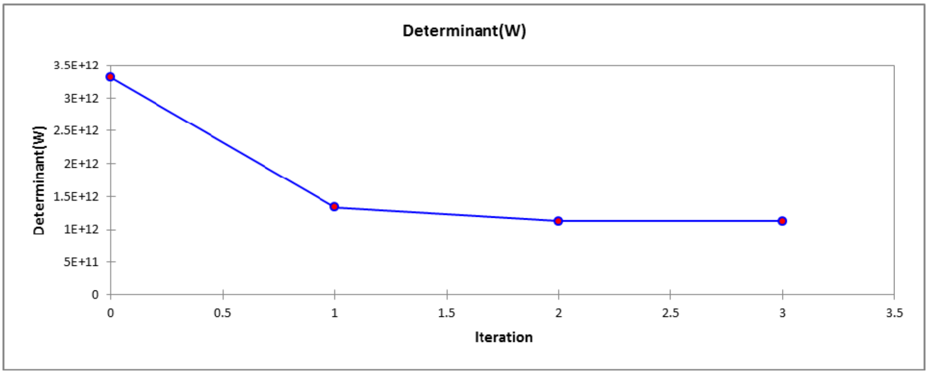


Fig. 2. Statistics for each iteration

Analysis of multiple runs of this experiment showed exciting results as well. Using 500 trials, it was discovered that, the algorithm guessed or classified requirement sets correctly. This is reflected in table 2, where the centroids for each variable were computed based on the stakeholder's weights. The sum of weights and variance for each requirement set was also calculated. The former aided in the prioritization of requirement sets, while the latter shows the variances existing between each requirement set.

Table 2. Class centroids

Class	Rate P	Point P	Rank P	Sum of weights	Within-class variance
1	4.604	17.347	0.276	53.00	7.302
2	4.230	7.6520	0.277	61.00	8.283
3	4.258	52.831	0.346	31.00	37.89
4	3.714	85.639	0.270	14.00	172.8
5	4.370	24.396	0.368	27.00	2.393
6	4.172	39.844	0.302	29.00	12.69
7	1.276	19.435	0.290	12.00	3.607
8	4.167	30.188	0.302	30.00	1.992
9	4.410	27.635	0.437	8.000	1.190
10	262.0	262.00	262.0	1.000	0.000

Table 3. Contribution (Analysis of variance)

Observation	DF(Model)	Mean squares (model)	DF (Error)	Mean square error	F	Pr > F
Rate P	1	733.847	264	249.847	2.937	0.088
Point P	1	82946.75	264	297.017	279.266	<0.0001
Rank P	1	774.132	264	255.557	3.029	0.083

Further analysis was performed using a two-way analysis of variance (ANOVA). On the overall dataset, we found significant correlations between the ranked requirements. The results of ANOVA shown in Table 3 produced significant effect on the Rate P and Rank P with minimized disagreement rates (p-value = 0.088 and 0.083 respectively). Also, the results of the ranked requirements are shown on the profile plot depicted in Figure 3. Our experiments generated 10 Gaussian clusters datasets as presented in Figure 1 and Table 2 respectively. Table 2 reflect the visual representations of the results, where the computed centroids were used in determining the relative ranks of generated clusters. The cluster shape, spread and spatial sizes are labelled according to variables specified during the experiment. Therefore, from Figure 3, it can be observed that Requirement set 4 was most ranked, followed by 3, 6, 9, 7, 1 in that order.

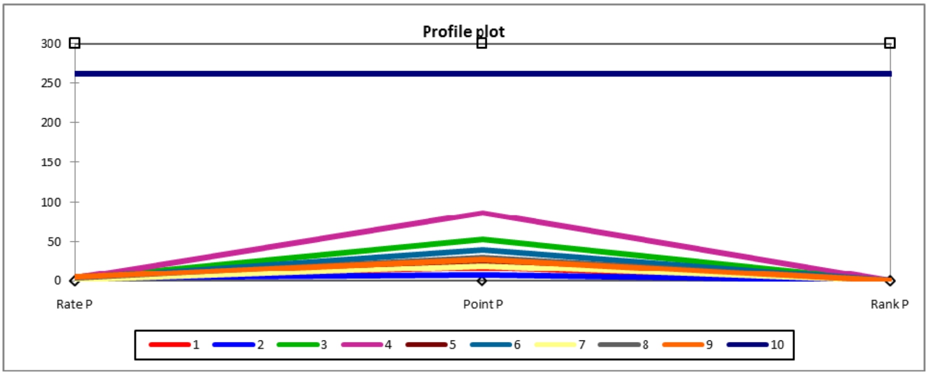


Fig. 3. Results by classes

6 Discussion

The aim of this research was to develop an enhanced prioritization technique based on the limitations of existing ones. It was eventually discovered that, existing techniques actually suffer from scalability problems, rank reversals, large disparity or disagreement rates between ranked weights as well as unreliable results. These were addressed at one point or the other during the course of undertaking this research. The method utilized in this research consisted of clustering algorithm with specific focus on k-means algorithm. Various algorithms and models were formulated in order to enhance the viability of the proposed technique. The evaluation of the proposed approach was executed with relevant datasets. The performance of the proposed technique was evaluated using ANOVA. The results showed high correlation between the mean weights which finally yielded the prioritized results. On the overall, the proposed technique performed better with respect to the evaluation criteria described in Section 4. It was also able to classify ranked requirements with the calculation of maximum, minimum and mean scores. This will help software engineers determined the most valued and least valued requirements which will aid in the planning for software releases in order to avoid breach of contracts, trusts or agreements. Based on the presented results, it will be appropriate to consider this research as an improvement in the field of computational intelligence.

7 Conclusion and Future Work

In conclusion, prioritizing requirements is an important aspect of software development process. In this paper, a clustering based technique has been proposed for prioritizing large number of requirements. This technique can help software engineers make qualitative decisions which include: (1) Requirement elicitation (2) setting criteria that constitute each requirement (3) envisioning the expected result or output (4) determining the weights of each criterion and (5) prioritizing the requirements. Most importantly, the ability to ensure objective selection or grading process will help in the quest to develop acceptable and robust software products. In our approach, the basic elements consist of criteria which define a specific requirement, ranked with weights which are combinations of numeric values. However, the benchmark of rank accuracy between the proposed and existing techniques is worth exploration. Also, in the future, we hope to develop a parallel hybridization of clustering and evolutionary algorithms to solve requirement prioritization problem.

Acknowledgement. This work is supported by the Research Management Centre (RMC) at the Universiti Teknologi Malaysia under Research University Grant (Q.J130000.2510.03H02), the Ministry of Science, Technology & Innovations Malaysia under Science Fund (R.J130000.7909.4S062) and the Ministry of Higher Education (MOHE) Under Exploratory Research Grant Scheme (R.J130000. 7828.4L051).

References

1. Perini, A., Susi, A., Avesani, P.: A machine learning approach to software requirements prioritization. IEEE Transactions on Software Engineering 39(4), 445–461 (2013)
2. Tonella, P., Susi, A., Palma, F.: Interactive requirements prioritization using a genetic algorithm. Information and Software Technology 55(1), 173–187 (2013)
3. Ahl, V.: An experimental comparison of five prioritization methods. Master's Thesis, School of Engineering, Blekinge Institute of Technology, Ronneby, Sweden (2005)

4. Berander, P., Andrews, A.: Requirements prioritization. In: Engineering and Managing Software Requirements, pp. 69–94. Springer, Heidelberg (2005)
5. Kobayashi, A., Maekawa, M.: Need-based requirements change management. In: Proceedings of the Eighth Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems, ECBS 2001, pp. 171–178. IEEE (2001)
6. Kassel, N.W., Malloy, B.A.: An approach to automate requirements elicitation and specification. In: International Conference on Software Engineering and Applications (2003)
7. Perini, A., Ricca, F., Susi, A.: Tool-supported requirements prioritization: Comparing the AHP and CBRank methods. Information and Software Technology 51(6), 1021–1032 (2009)
8. Racheva, Z., Daneva, M., Herrmann, A., Wieringa, R.J.: A conceptual model and process for client-driven agile requirements prioritization. In: 2010 Fourth International Conference on Research Challenges in Information Science (RCIS), pp. 287–298. IEEE (2010)
9. Berander, P., Khan, K.A., Lehtola, L.: Towards a research framework on requirements prioritization. SERPS 6, 18–19 (2006)
10. Achimugu, P., Selamat, A., Ibrahim, R., Mahrin, M.N.R.: A systematic literature review of software requirements prioritization research. Information and Software Technology (2014)
11. Kaur, J., Gupta, S., Kundra, S.: A kmeans clustering based approach for evaluation of success of software reuse. In: Proceedings of International Conference on Intelligent Computational Systems, ICICS 2011 (2011)
12. Lim, S.L., Finkelstein, A.: StakeRare: using social networks and collaborative filtering for large-scale requirements elicitation. IEEE Transactions on Software Engineering 38(3), 707–735 (2012)
13. Lim, S.L., Harman, M., Susi, A.: Using Genetic Algorithms to Search for Key Stakeholders in Large-Scale Software Projects. In: Aligning Enterprise, System, and Software Architectures, pp. 118–134 (2013)

A Comparative Evaluation of State-of-the-Art Cloud Migration Optimization Approaches

Abdelzahir Abdelmaboud, Dayang N.A. Jawawi, Imran Ghani, and Abubakar Elsafi

Department of Software Engineering, Faculty of Computing,
Universiti Teknologi Malaysia, 81310 UTM,
Skudai, Johor, Malaysia
{abdzahir,bakri1985}@hotmail.com, {dayang,imran}@utm.my

Abstract. Cloud computing has become more attractive for consumers to migrate their applications to the cloud environment. However, because of huge cloud environments, application customers and providers face the problem of how to assess and make decisions to choose appropriate service providers for migrating their applications to the cloud. Many approaches have investigated how to address this problem. In this paper we classify these approaches into non-evolutionary cloud migration optimization approaches and evolutionary cloud migration optimization approaches. Criteria including cost, QoS, elasticity and degree of migration optimization have been used to compare the approaches. Analysis of the results of comparative evaluations shows that a Multi-Objectives optimization approach provides a better solution to support decision making to migrate an application to the cloud environment based on the significant proposed criteria. The classification of the investigated approaches will help practitioners and researchers to deliver and build solid approaches.

Keywords: Cloud computing, application migration, optimization, evolutionary algorithms.

1 Introduction

Cloud computing is a new computing technology paradigm, due to the rapidly increasing progress of IT technologies that have made cloud computing a significant research topic in information technology and scientific research [1],[2],[3]. Cloud computing offers shared services, information, storage resources and computing to users across the internet on request. It provides many features to consumers, including low operational cost and high reliability of the application. As a result, cloud computing has been widely used in e-business, education and scientific research, etc. Cloud services can be classified fundamentally into three classes; Software as a Service (SaaS), which offers access to applications and systems to consumers [4]. Platform as a Service (PaaS), which provides a computing platform to application developers to enable them to design, develop, deploy and test activities [3], [4]. Infrastructures as a Service (IaaS) offers shared resources and storage to consumers [4].

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_60, © Springer International Publishing Switzerland 2014

Cloud application migration can be defined as moving an application from a local platform to a service provider cloud environment. To migrate an application to the cloud environment requires many processes: migration assessment, architecture and planning, proof of concepts (validated approach), migration (data, application and process) and optimization and testing. Cloud migration optimization is activity during or after the test migration process to enhance performance and resource efficiency as well as reduce the cost of cloud application migration.

Nowadays, cloud computing has attracted increasing interest from both industry and academic research and there is massive demand for leveraging existing systems of cloud computing technologies. However, there are still challenges to migrate and deploy enterprise software and applications to the cloud [5],[6]. Cloud computing can be easily utilized to develop enterprise software in a constitutive project. Extensive re-engineering activities are required to run enterprise software and applications on cloud computing during migration, instead of rebuilding software systems and applications from scratch [7].

The major benefits of cloud applications migration allow an application provider (SaaS provider) to reuse intrinsic components of a system instead of building software applications from scratch that are compatible with cloud environments. However, there are diverse primary obstacles that impede the migration of applications. Current approaches do not support automatic migration for the cloud environment and very limited to particular cloud environments [7]. Furthermore, there are many combinations of options involved in migrating and deploying an application to the cloud. These options vary broadly in their performance and characteristics such as mixing various resources (CPU, storage, memory and network options), and multiple approaches to enable scalability of dynamic resources [8].

Moreover, to migrate an application to the cloud environment requires many aspects but in this study we focus non-functional properties such as cost and QoS in terms of response time Service Level agreement (SLA) violation. The cost and QoS need to be optimized during migration to allocate the amount of resources required; this is known as NP-hard and may conflict with cost and QoS objectives [9], [10].

In this paper, we report on a recent survey of Cloud Migration Optimization (CMO) approaches and problems. These approaches have been contrasted with respect to some of the criteria. This study provides an overview and categorization of the proposed CMO approaches that can help practitioners and researchers to deliver and build solid approaches.

The structure of this paper is as follows. Section 2 discusses the classification of the CMO approaches. Section 3 provides the comparative evaluation with some criteria. Section 4 describes the results of the comparative evaluation and challenges. Finally, the conclusion of this work is given in Section 5.

2 Classification of the CMO Approaches

The CMO approaches have been classified into two main approaches. non-evolutionary cloud migration optimization (NCMO) and evolutionary cloud migration optimization (ECMO) approaches.

Non-evolutionary or classical optimization approaches are used to achieve a single optimal solution in one simulation run. In solving Multi-Objectives optimization problems, they need to be repeated many times to find various optimal solutions every run time. However, this type of solution is not useful for users and it is not satisfactory to make decisions by achieving an optimal solution with frequent to single criteria [11]. For these reasons, we classify NCMO approaches based on literature that proposes solutions such as architecture, model and tool approaches rather than single and Multi-Objectives optimization categorization.

On the other hand, ECMO approaches based on evolutionary algorithms offer the best choice of Multi-Objectives optimization by finding optimal solutions (Pareto-optimal) to various objectives. That is because they deal with solutions involving population of their search spaces that provide Pareto-optimal solutions in only a single run. Moreover, evolutionary algorithm approaches provide simultaneous among multiple conflicting objectives [11]. Therefore, our categorization of ECMO approaches depends on single and multiple objectives rather than previous classification of the classical optimization approaches such as architecture and model approaches. These are shown in Fig. 1. and described in the following sections:

2.1 NCMO Approaches

Most non-evolutionary cloud migration optimization (NCMO) approaches or classical optimization approaches address cost as a significant factor for businesses related to consumers and service providers as regards cloud services. These approaches can be categorized into sub-approaches as follows.

Architecture-Based Approach. An architecture-based approach needs to be adaptive during transformation runtime migration in order to support the move of applications or software systems to cloud environments. However, very limited architecture-based approaches have been proposed in the literature relevant to cloud migration optimization. For example, Liu et al. [1] propose an architecture approach of optimization service deployment to reduce costs, improve efficiency of deployment and guarantee the consumers' QoS requirements. In particular, they propose three algorithms to standardize and optimize the requirements of service deployment. The proposed approaches have been evaluated using a simulation experiment and shown to be effective and efficient. Meanwhile, Frey et al. [12] have presented an architecture for detecting and checking a violation of software systems by considering migration support. The proposed approach helps to actively highlight and detect important system parts at an early stage. Chen et al. [13], on the other hand, present an architecture approach for optimizing QoS in respect of Dynamic Data Driven application systems. However, the proposed approach lacks validation and evaluation through experiments.

Model-Based Approach. Most model-based approaches propose in first step of migration assessment focusing on cost and performance in terms of response time of cloud migration optimization with respect to application resources management and cost analysis. Ghanbari et al. [14] have suggested a two models approach relevant to QoS level of application resources and maps of service level consumption of resource to profit metrics. The focus of the proposed approach is to solve optimization problems of resources allocation in a private cloud with regard to minimized costs through maximized sharing of resources. The results of the proposed approach show that optimization is accurate and profit is increased by reducing costs. Similarly, a two models approach has been presented for Enterprise Resource Planning (ERP-SAP) with regard to service level agreements [15]. The first model is a queue network for application performance. The second model is cost analysis for fixed costs of hardware and costs of dynamic operation. The approach presented shows efficient use by service providers for planning decisions with respect to SLA.

Frey et al. [7] present a model-based approach (CloudMIG) targeting semi-automatic migration of software systems with regard to resource efficiency and scalability of IaaS and PaaS-based applications. The proposed approach experiments provide initial categorization of cloud compatible but not offer any improvement in cloud migration optimization. Li et al. [16], meanwhile, have presented a performance model-based approach to develop, deploy and operate cloud applications. The proposed approach aims to maximize profit with QoS and SLA through multiple workloads of cloud applications.

Tool-Based Approach. The tool-based approach is very important to use for validating cloud migration in general. In particular, it helps consumers, application providers and service providers to make decisions before applications are deployed in the cloud. However, it lacks tools supporting migration to the cloud and its challenges require more research [17]. Very limited tool-based approaches are found in the literature, such as Ferrer et al. [18], who present a toolkit (OPTIMIS) to optimize the life cycle of service of cloud infrastructure that includes service construction, deployment and operation with regard to some aspects of trust, efficiency, risk and cost. The proposed tools enable developers to improve services with non-functional requirements with regard to consumption, trust, and cost. In addition, the tool enhances decision making to select appropriate service providers and infrastructure providers.

Fittkau et al. [19] have presented a tool approach known as "CDOSIM" for simulating cost and performance in terms of response time of cloud deployment options to support software systems migration. The results of the proposed approach are accurate prediction of performance (response time) and cost compared with Amazon EC2 and Eucalyptus.

CloudGenius is a tool-based approach that supports decision making of the web application migration to the cloud. In particular, the CloudGenius approach solves the problem of web applications to cloud virtualization service to select suitable software images and services of infrastructure to enable QoS to achieve applications targets [20].

2.2 Evolutionary Cloud Migration Optimization (ECMO) Approaches

Search-Based Software Engineering (SBSE) has become commonplace and well known in Software Engineering. The goal of SBSE is to build automation solutions to the Software Engineering problems based on optimization algorithms approaches [21]. SBSE offers great opportunities to evaluate research and solve optimization problems, for instance, to optimize QoS objectives for cloud migration systems [22]. Furthermore, evolutionary algorithms are what most SBSE approaches use to solve optimization problems of cloud migration because they are easily parallelized and highly scalable [23], [24].

ECMO approaches in migration applications to the cloud can be classified into Single Objective Optimization (SOO) approach and Multi-Objectives Optimization (MOO) approach, and described as follows.

Single Objective Optimization (SOO) Approach. The SOO approach refers to one quality attribute. There are very limited SOO approaches to be found in the literature related to ECMO approaches in the cloud. For example, Pandey et al. [25] propose an approach using particle swarm optimization (PSO) to reduce the execution costs of application workflows in the cloud. The results of the proposed approach provide cost savings which are three times better compared with "Best Resource Selection". Meanwhile, Csorba et al. [26] have proposed a Colony Optimization (CO) approach for deployment of virtual machines (VMs) images onto physical machines. The proposed approach improves the scalability of systems.

Multi-Objectives Optimization (MOO) Approach. The MOO approach refers to two or more quality attributes obtained simultaneously. These quality attributes provide a set of solutions that are trade-off of Pareto-optimal. However, few research projects have been completed on MOO approaches and they are still in the initial stage of SBSE.

Frey et al. [5] propose an approach in respect of the genetic algorithm (CDO Xplorer) of deployment optimization options. The proposed approach targets enhancing multi-optimization in terms of cost, response time and SLA violation to support software systems migration in the cloud. The demonstrated results of the approach show a better solution, up to 60% compared with experiments in Microsoft Windows Azure and Amazon EC 2 cloud environments.

Wada et al. [10] have presented a genetic algorithm (E3 –R) approach that searches Pareto-optimal sets solutions of deployment configurations to satisfy SLAs and QoS objectives confliction. The approach demonstrates efficient satisfaction of SLAs in a short time and achieves quality deployment service configurations.

Yusoh and Maolin [27] have presented two approaches. The first approach is a Cooperative Coevolution Genetic Algorithm (CCGA) for initial placement of SaaS problems. The second approach is a Repair-based Group Genetic Algorithm for resource optimization of SaaS problems. Their experiment showed that evolutionary algorithms provide better efficiency and scalability.

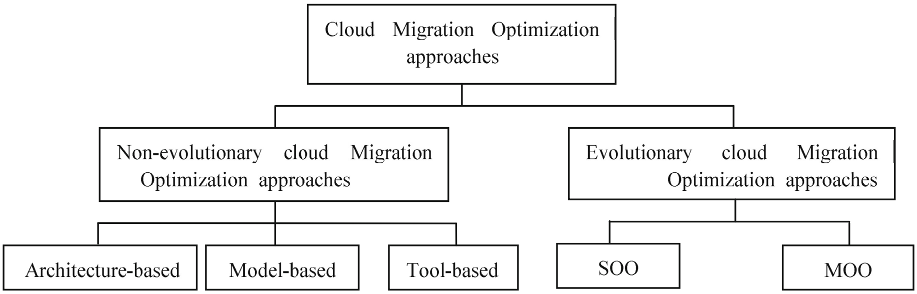

Cloud Migration Optimization approaches
Non-evolutionary cloud Migration Optimization approaches
Evolutionary cloud Migration Optimization approaches
Architecture-based
Model-based
Tool-based
SOO
MOO

Fig. 1. Classification of cloud migration optimization approaches

3 Comparative Evaluation

This section explains the criteria used to evaluate the CMO approaches set out above. These criteria satisfy any approaches to CMO that are commonly used in the literature [28], [29] ,[30],[31].

Cost. The attractions and new trends of an organization to use cloud resources instead of those in data centres can save costs related to economies of scale, because the costs of power, hardware and administrative support are about five times better. Moreover, if an organization's business grows, the cloud offers elasticity of costs rather than having to purchase expensive additional resources when a system require more capacity [30].

The most important criterion for consumers and service providers in making decisions on cloud application migration and support is the cost. In particular, Brebner and Liu [32] have reported on running applications costs with multiple workloads in different Amazon, Microsoft and Google offering when choosing suitable cloud service providers for applications migration.

QoS. A Service Level Agreement (SLA) is a contract agreed between consumers and service providers as the first stage to enable migration of cloud applications or systems to cloud environments. This SLA is specified by cost and QoS parameters. As a result, QoS is an important criterion and a critical issue in migrating applications to the cloud. The most important QoS criteria provided by service providers are performance (response time, throughput and latency), SLA violation, availability and reliability [29]. In this study we focus on performance of the application migration regarding response time and SLA violation, because they are most significant to consumers and service providers. They are defined as follows.

- **Response Time.** The consumer will get the right services that they need at specific times.
- **SLA Violation.** ("Indicate the number of method calls with response time that exceed a given time").

Elasticity. Elasticity is the ability for a customer to quickly request, receive, and release as many resources as needed [29]. Elasticity is also defined as the degree of scalability that offers the means for optimizing usage of resources in situations of wiggle or/and unknown application workloads [28]. Elasticity can be classified into groups [29].

- **Horizontal Scalability.** depends on applications instances or components offer and enable adding more applications instances when needed.
- **Vertical Scalability.** depends on service providers that offer an approach to scaling resources to applications.

Degree of Migration Automation. Application migration automation is challenged by the growing number of systems and allows different ways to adapt IT systems within control. Automated approaches to service migration and deployment and configuration prevent human errors and make the process easier. The degree of automation is categorized into three classes as follows [17].

- **Manual.** ("A manual approach which should be performed by a human being").
- **Semi-automated.** ("A solution that is partially automated by software tools").
- **Full automated.** ("A fully automated migration, whether it is a model transformation or decision support").

4 Results of the Comparative Evaluation and Challenges

This section describes the comparative evaluation results of the two main approaches. The first approach - NCMO - consists of three sub-approaches. architecture-based, model-based and tool-based approaches, while the second approach - ECMO - comprises two sub-approaches. SOO and MOO approaches. We have used the indicators low, average and high to explain an approach level regarding each proposed criteria.

Cost. All NCMO approaches show high scores for cost because of the importance of the cost criterion for consumers making decisions on the migration of systems to the cloud. Cost is a critical factor for consumers to choose appropriate service providers that offer services at a reasonable price and with a high QoS. However, there is a need for systematic architecture to adapt systems in order to support migration to the cloud [17].

On other side, in ECMO approaches, the SOO approach has average scores while MOO has high scores for cost to support migrations of systems. However, the cost is significant for businesses looking to make decisions, and a challenge which requires a cost-benefit analysis to migrate an application to the cloud environment [30].

QoS. Architecture-based approach scores are low for QoS considering performance. The QoS criterion for the architecture-based approach remains cloud challenging due to various applications migration requirements between traditional hosting and centralized cloud infrastructure.

Model-based approach scores are high for QoS in respect of performance support migration because most of the proposed solutions are models to optimize applications performance on migrating to the cloud; for example, Li et al. [15] present the queue network model for application performance. Whereas, tool-based approaches scores are average for QoS considering performance for migrating application optimization; for instance, Fittkau et al. [19] propose a tool for optimizing migration of applications performance. However, as mentioned before in section 3, there is still a lack of tools to support applications migrations into the cloud infrastructure, and that requires more research investigation.

In ECMO approaches, the SOO approach has average scores while the MOO approach has high scores for QoS (performance) criteria. However, evolutionary algorithms are at an early stage, and remain a promising research area and challenge for Software Engineering researchers. Further research is needed in Software Engineering in general in the future. In particular, more research is required on the re-engineering of systems to be suitable for cloud migration and how to minimize costs by considering optimization resource usage [24].

All CMO approaches have low scores with regard to SLA violation of QoS criteria except tool-based scores, which are average. However, there is a critical need for SLA violation prevention in order to avoid penalties that service providers have to pay to consumers [33].

Elasticity. All approaches of NCMO scores are low, whereas SOO approach scores are average and MOO approach scores are high for elasticity criteria. Elasticity is a serious factor of cloud deployment in business and engineering challenges to cloud consumers because elasticity reduces costs by optimizing resources usage [24].

Most CMO approaches support manual and semi-automated migration systems to the cloud. However, only one study [5] is based on a MOO approach that supports full automation migration systems and provides an open source tool (CloudMIG) for Multi-Objectives genetic algorithm optimization in terms of cost, performance and SLA violation. It is clear that there is a lack of full automation tools to support systems migrations to the cloud [17].

In summary, the MOO approach of ECMO is highly suitable to solve and achieve the proposed study criteria of cost, QoS, elasticity and degree of migration automation compared with SOO and NCMO approaches. Moreover, the MOO approach provides a better solution of Multi-Objectives optimization and supports decision making better than the SOO and NCMO approaches for many reasons [10].

- The SOO approach cannot optimize trade-offs through simultaneous conflicting cost and QoS objectives.
- The NCMO approach cannot achieve cost and QoS objectives at once within trade-offs.
- The NCMO approach does not make it easy to define a problem in the linear programming form.
- The NCMO approach is not scalable and parallelized.

The summary results of the comparative evaluation between the five sub-approaches are shown in Table 1.

Table 1. Summary result of the comparative evaluation

Criteria	Cost	QoS		Elasticity	Degree of migration automation			Overall result
Approach		Response time	SLA violation		Manual	Semi-automation	Full-automation	
Architecture-based	High	Low	Low	Low	√	√	×	Low
Model-based	High	High	Low	Low	√	√	×	Average
Tool-based	High	Average	Average	Low	√	√	×	Average
SOO	Average	Low	Low	Average	×	×	×	Low/ Average
MOO	High	High	Low	High	√	√	√	High

4.1 The Proposed Framework

Our proposed solution is to develop a Multi-Objectives optimization framework as an approach to support for making decisions to migrate an application to the cloud environment. Based on the evaluation results above, MOO approach has been selected to implement the proposed framework. In particular, search space for suitable solutions due to different combinations options requirements of the application and multiple cloud environments that offer different resources with various costs and different performance characteristics are a huge and critical issue. For example, a cloud user needs to find the best solutions among cost, response time, SLA violation objectives known as trade-offs Pareto optimum, i.e. to achieve a single objective - reduce cost - may delay the performance objective. This does not improve the optimal solution. In particular, these guides to the following research challenges:

- How application providers to estimate the optimal solutions to migrate their customer applications to cloud environments considering to cost and QoS in terms of performance (response time) and SLA violation?
- How application providers decide to select best service provider regarding customer requirements of cost and QoS to migrate the applications to cloud environments?
- What is a trade-off between cost and QoS to migrate the applications to cloud environments?

The proposed framework will help application customer and provider to assess and make the decision to select an appropriate cloud provider. The framework will provide trade-offs optimal solutions of multi-objectives optimization problems of cost and QoS (response time and SLA violation) to migrate an application to the cloud environment by proposing to use many features and components to the framework as shown in Fig. 2. And described as follows.

Application Customer. Application customer is the user or organization that required to migrate an application cloud environments.

Application Provider. Application provider or SaaS provider is a special kind of cloud user that provides SaaS services to cloud SaaS users or customers among web applications.

Service Provider. Service provider (SP) offers resources and services to cloud users based-on PaaS and IaaS.

CloudMIG. We want to use a CloudMIG tool simulator that provides the following features among other tools.

- Open source code tool available for future research investigation.
- Support full automation of cloud applications migration based on IaaS of clouds owing to lack of automation tools to support cloud applications migration.
- Enable to assess and evaluate among huge search-based space in order to achieve Multi-Objectives optimization of migrating applications.
- Support simulation of main cloud vendors' platforms or service providers as cloud profiles to help in assessment and create realistic scenarios.

Differential Evolution Algorithm. We will use a differential evolution (DE) algorithm to achieve Multi-Objectives optimization of cost, response time and SLA violation. Using a DE algorithm will provide better outcomes than the genetic algorithm (GA) (CDOXplorer) [5] that used the same CloudMIG tool of cloud migration optimization for many reasons; DE is more efficient than GA in exploring the decision space of Multi-Objectives optimization [34] ,[35], and DE offers more accuracy and stability and better convergence speed than GA for solving various optimization problems of applications [36], [37].

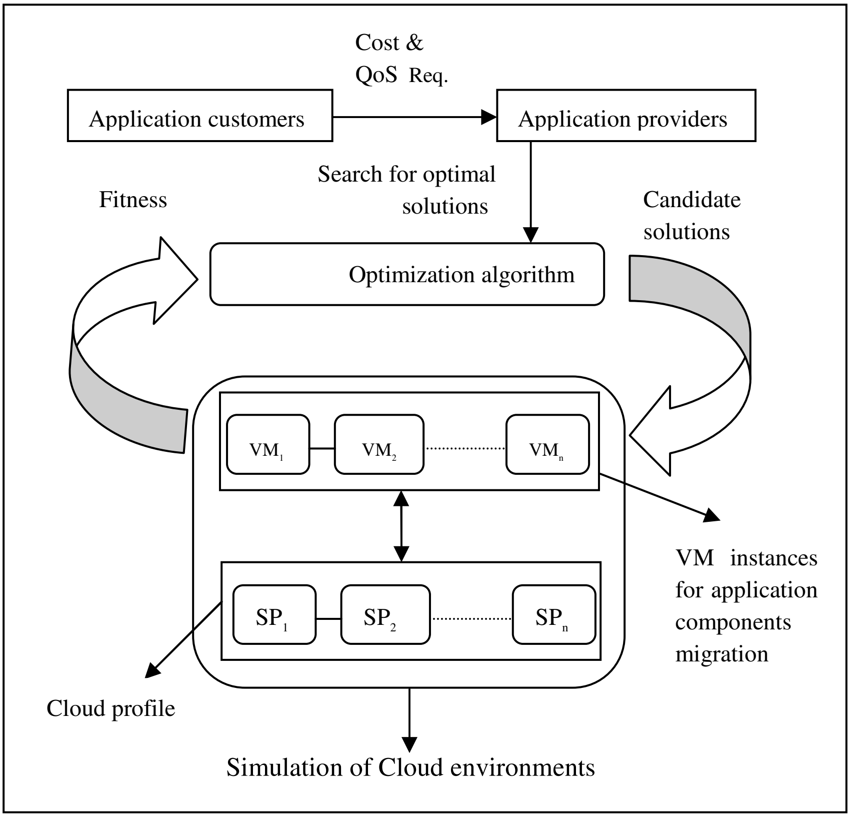


Fig. 2. General view of the proposed framework

5 Conclusion

Our aim of this study is to provide an overview and contrast, recent research in cloud migration optimization. We have categorized these two main approaches. NCMO approaches have been classified into architecture-based, model-based and tool-based approaches, while ECMO approaches have been categorized into SOO and MOO approaches. We proposed four important criteria - cost, QoS, elasticity and degree of migration automation - for these approaches. Based on analysis of the results of comparative evaluation, we found that the MOO approach scored higher than the model-based, architecture-based, tool-based and SOO approaches and provided a better solution to support decision making on Multi-Objectives optimization of application migration to a cloud environment.

While surveying these approaches, many challenges were found that require more research attention in the future, such as the lack of a systematic architecture approach at runtime to support adoption during migration of an application to the cloud. There is a need for tools to evaluate and support application migration to the cloud environment, and a lack of full automation tool to support applications migration to the cloud.

Our future work will aim to conclude the proposed framework that used the CloudMIG simulation and DE algorithms to support decision making and achieve Multi-Objectives optimization of cloud application migration.

Acknowledgements. Special thanks to Universiti Teknologi Malaysia (UTM) and to the Ministry of Science, Technology and Innovation Malaysia (MOSTI) for financing and funding this research through escience fund with number Vote No.4S064. Also, to our Embedded & Real-Time Software Engineering Laboratory (EReTSEL) members for their continuous support.

References

1. Liu, T., Lu, T., Wang, W., Wang, Q., Liu, Z., Gu, N., Ding, X.: SDMS-O: A service deployment management system for optimization in clouds while guaranteeing users' QoS requirements. Future Gener. Comput. Syst. 28, 1100–1109 (2012)
2. Buyya, R., Yeo, C.S., Venugopal, S., Broberg, J., Brandic, I.: Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation Computer Systems 25, 599–616 (2009)
3. Marios, D.D., Dimitrios, K., Pankaj, M., George, P., Athena, V.: Cloud Computing: Distributed Internet Computing for IT and Scientific Research. In: Dimitrios, K., Pankaj, M., George, P., Athena, V. (eds.) IEEE Internet Computing, vol. 13, pp. 10–13 (2009)
4. Wikipedia, http://en.wikipedia.org/wiki/Cloud_computing
5. Frey, S., Fittkau, F., Hasselbring, W.: Search-based genetic optimization for deployment and reconfiguration of software in the cloud. In: Proceedings of the 2013 International Conference on Software Engineering, pp. 512–521. IEEE Press, San Francisco (2013)
6. Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J., Ghalsasi, A.: Cloud computing — The business perspective. Decision Support Systems 51, 176–189 (2011)
7. Frey, S., Hasselbring, W.: The CloudMIG Approach: Model-Based Migration of Software Systems to Cloud-Optimized Applications. International Journal on Advances in Software, 342–353 (2011)
8. Grundy, J., Kaefer, G., Keong, J., Liu, A.: Guest Editors' Introduction: Software Engineering for the Cloud. IEEE Software 29, 26–29 (2012)
9. Canfora, G., Penta, M.D., Esposito, R., Villani, M.L.: An approach for QoS-aware service composition based on genetic algorithms. In: Proceedings of the 2005 Conference on Genetic and Evolutionary Computation, pp. 1069–1075. ACM, Washington DC (2005)
10. Wada, H., Suzuki, J., Yamano, Y., Oba, K.: Evolutionary deployment optimization for service-oriented clouds. Softw. Pract. Exper. 41, 469–493 (2011)
11. Ghosh, A.: Evolutionary algorithms for multi-criterion optimization: a survey. International Journal of Computer & Information Sciences (2004)

12. Frey, S., Hasselbring, W.: An Extensible Architecture for Detecting Violations of a Cloud Environment's Constraints during Legacy Software System Migration. In: 15th European Conference on Software Maintenance and Reengineering (CSMR), pp. 269–278 (2011)
13. Chen, T., Bahsoon, R., Theodoropoulos, G.: Dynamic QoS Optimization Architecture for Cloud-based DDDAS. Procedia Computer Science 18, 1881–1890 (2013)
14. Ghanbari, H., Simmons, B., Litoiu, M., Iszlai, G.: Feedback-based optimization of a private cloud. Future Generation Computer Systems 28, 104–111 (2012)
15. Li, H., Casale, G., Ellahi, T.: SLA-driven planning and optimization of enterprise applications. In: Proceedings of the First Joint WOSP/SIPEW International Conference on Performance Engineering, pp. 117–128. ACM, San Jose (2010)
16. Li, J., Chinneck, J., Woodside, M., Litoiu, M., Iszlai, G.: Performance model driven QoS guarantees and optimization in clouds. In: Proceedings of the 2009 ICSE Workshop on Software Engineering Challenges of Cloud Computing, pp. 15–22. IEEE Computer Society (2009)
17. Pooyan, J.: Cloud Migration Research: A Systematic Review. IEEE Transactions on Cloud Computing 99, 1 (2013)
18. Ferrer, A.J., Hernández, F., Tordsson, J., Elmroth, E., Ali-Eldin, A., Zsigri, C., Sirvent, R., Guitart, J., Badia, R.M., Djemame, K., Ziegler, W., Dimitrakos, T., Nair, S.K., Kousiouris, G., Konstanteli, K., Varvarigou, T., Hudzia, B., Kipp, A., Wesner, S., Corrales, M., Forgó, N., Sharif, T., Sheridan, C.: OPTIMIS: A holistic approach to cloud service provisioning. Future Generation Computer Systems 28, 66–77 (2012)
19. Fittkau, F., Frey, S., Hasselbring, W.: CDOSim: Simulating cloud deployment options for software migration support. In: IEEE 6th International Workshop on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA), pp. 37–46 (2012)
20. Menzel, M., Ranjan, R.: CloudGenius: Decision Support for Web Server Cloud Migration. In: Proceedings of the 21st International Conference on World Wide Web. eprint arXiv:1203.3997 (2012)
21. Harman, M.: The Current State and Future of Search Based Software Engineering. In: Future of Software Engineering, FOSE 2007, pp. 342–357 (2007)
22. White, D.R.: Cloud Computing and SBSE. In: Ruhe, G., Zhang, Y. (eds.) SSBSE 2013. LNCS, vol. 8084, pp. 16–18. Springer, Heidelberg (2013)
23. Harman, M.: Software Engineering Meets Evolutionary Computation. Computer 44, 31–39 (2011)
24. Harman, M., Lakhotia, K., Singer, J., White, D.R., Yoo, S.: Cloud engineering is Search Based Software Engineering too. Journal of Systems and Software 86, 2225–2241 (2013)
25. Pandey, S., Linlin, W., Guru, S.M., Buyya, R.: A Particle Swarm Optimization-Based Heuristic for Scheduling Workflow Applications in Cloud Computing Environments. In: 24th IEEE International Conference on Advanced Information Networking and Applications (AINA), pp. 400–407 (2010)
26. Csorba, M.J., Meling, H., Heegaard, P.E.: Ant system for service deployment in private and public clouds. In: Proceedings of the 2nd Workshop on Bio-Inspired Algorithms for Distributed Systems, pp. 19–28. ACM, Washington, DC (2010)
27. Yusoh, Z.I.M., Maolin, T.: Composite SaaS Placement and Resource Optimization in Cloud Computing Using Evolutionary Algorithms. In: IEEE 5th International Conference on Cloud Computing (CLOUD), pp. 590–597 (2012)
28. Andrikopoulos, V., Binz, T., Leymann, F., Strauch, S.: How to adapt applications for the Cloud environment. Computing 95, 493–535 (2013)
29. Badger, M.L., Grance, T., Patt-Corner, R., Voas, J.M.: Cloud Computing Synopsis and Recommendations. NIST Special (2012)

30. Tran, V., Keung, J., Liu, A., Fekete, A.: Application migration to cloud: a taxonomy of critical factors. In: Proceedings of the 2nd International Workshop on Software Engineering for Cloud Computing, pp. 22–28. ACM Press, Waikiki (2011)
31. Andrikopoulos, V., Strauch, S., Leymann, F.: Decision Support for Application Migration to the Cloud: Challenges and Vision. In: Proceedings of the 3rd International Conference on Cloud Computing and Service Science, pp. 149–155. SciTePress (2013)
32. Brebner, P., Liu, A.: Performance and Cost Assessment of Cloud Services. In: Maximilien, E.M., Rossi, G., Yuan, S.-T., Ludwig, H., Fantinato, M. (eds.) ICSOC 2010. LNCS, vol. 6568, pp. 39–50. Springer, Heidelberg (2011)
33. Emeakaroha, V.C., Netto, M.A.S., Calheiros, R.N., Brandic, I., Buyya, R., De Rose, C.A.F.: Towards autonomic detection of SLA violations in Cloud infrastructures. Future Generation Computer Systems 28, 1017–1029 (2012)
34. Tušar, T., Filipič, B.: Differential Evolution versus Genetic Algorithms in Multiobjective Optimization. In: Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T., Murata, T. (eds.) EMO 2007. LNCS, vol. 4403, pp. 257–271. Springer, Heidelberg (2007)
35. dos Santos Amorim, E.P., Xavier, C.R., Campos, R.S., dos Santos, R.W.: Comparison between Genetic Algorithms and Differential Evolution for Solving the History Matching Problem. In: Murgante, B., Gervasi, O., Misra, S., Nedjah, N., Rocha, A.M.A.C., Taniar, D., Apduhan, B.O. (eds.) ICCSA 2012, Part I. LNCS, vol. 7333, pp. 635–648. Springer, Heidelberg (2012)
36. Dong, X.-L., Liu, S.-Q., Tao, T., Li, S.-P., Xin, K.-L.: A comparative study of differential evolution and genetic algorithms for optimizing the design of water distribution systems. J. Zhejiang Univ. Sci. A 13, 674–686 (2012)
37. Das, S., Suganthan, P.N.: Differential Evolution: A Survey of the State-of-the-Art. IEEE Transactions on Evolutionary Computation 15, 4–31 (2011)

A Review of Intelligent Methods for Pre-fetching in Cloud Computing Environment

Nur Syahela Hussien, Sarina Sulaiman, and Siti Mariyam Shamsuddin

Soft Computing Research Group Faculty of Computing
Universiti Teknologi Malaysia, Skudai Johor, Malaysia
nursyahela90_@yahoo.com, {sarina,mariyam}@utm.my

Abstract. Innovation of technology in our world has expanded drastically. People like to use the applications that can easier their work in everyday's life. They have many data to be store and today the people like to store their data in cloud computing storage because the can access the data everywhere at anytime. Besides, most of users of smart phone access their data from storage that are outside from their mobile phone. This trend called as Mobile Cloud Computing and it changes the way users use the computer and the Internet. Even though, by increasing the number of users accesses the storage, it slows down the performance service of cloud computing. Due to these issues, the current researchers have applied a pre-fetching method as one of the method to improve on performance services. However, there are some limitations on pre-fetching method, which is the overhead that is cause by overaggressive pre-fetching. Therefore, in this paper a review on the ideas of enhancing the accessibility of Cloud Computing is explore by using the intelligent methods to improve the current pre-fetching method.

Keywords: Intelligent methods, mobile cloud computing, pre-fetching.

1 Introduction

Cloud Computing (CC) provides computing resources for their users through the Internet that acts as virtual computing resource. It can deploy, share out or change around a computing resource robustly and control the usage of resources every time [1]. When there are a lot of users use request the same data at the same time, the latency will occurs and the users need to wait for a while for access the data. Besides, when the users have kinds of cloud storage, it will take times for the users to recall back which cloud storage they store their current data. This issue is related to management problem and slow down the performance service of the cloud. Hence, it needs some improvements by applying some methods to maintain the quality of cloud service.

One of the best techniques to enhance the web performance is web caching and pre-fetching by keeping the web objects that are expect for future visit closer to the client. Web caching can work alone or integrate with web pre-fetching. The Web caching and pre-fetching can synchronise between each other because the

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_61, © Springer International Publishing Switzerland 2014

web-caching uses the temporal area to forecast revisiting requested objects, while web pre-fetching predicts web objects that might be requested in the near future, as the users have yet to request for these objects. Then, the predicted objects fetched from the origin server are stored in a cache. Thus, web pre-fetching helps in increasing the cache hits and reducing the user-perceived latency [2].

Thus, in order to understand the challenges and provide further scope for this research, comprehension of this novel approach is significant. This paper introduces the basic knowledge of CC, the current researchers, and the proposed method to handle the issues in perspective. The paper organized as section 1 introduces the technology of CC and the common method in handling the latency issues including the caching and pre-fetching. Section 2 gives the background study and the definitions of CC that are being important for the people today's. Section 3 discusses on enhance the pre-fetching method by using intelligent methods. The conclusions clarified in Section 4.

2 Cloud Computing

CC is the technology that is used by the Internet and servers to store data and applications [3]. It allows users to use applications or tools without install the software and they may access their data using any device at any time with Internet access. By centralising the data storage, processing, and bandwidth, it allows for much more efficient computing with this technology. CC is becoming an object of interest for both the publications and among users, including the individuals at home. It also allows users to obtain network storage space and resources of the computer using subscription-based service [4].

The CC is use by Salesforce.com since 1999 that introduced the concept of delivering enterprise application through a basic website. Next, in 2002, Amazon Web Service has launched. Then, Google Doc in 2006, and Eucalyptus in 2008 followed it, which is the first open source for private clouds. Microsoft has entered into CC in 2009. The latest to enter into CC are Oracle, Dell, Fujitsu, Teradata and HP [5].

Applications that are conveyed as services via the Internet are known as CC [6]. Mobile CC usually is use to run an application for instance, the Google's Gmail, on a remote resource rich server. One of the reasons for people to like using mobile cloud is that they can store and access their personal data on mobile device cloud [6]. Therefore, users can access the data easily, anywhere and at any time. CC are usually classified into three main service classes, which are Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) [3] [7] [8]. Along with Lauchlan (2012) conducted a survey for the future of CC and the result shows that about 82% of the users use SaaS for their businesses today and 84% are using SaaS as the deployment model for new applications [9] [10]. According to Lauchlan [9] SaaS is expected to be used by 88% users in five year starting now. Besides, Gens [11] predicts that in the next 3 years, SaaS vendors will grab the software market leadership. In addition, 2013 would be the year of brilliance for business clouds in small and medium sizes of usage. Therefore, with the increase in the number of demand for SaaS, its usage will lead to some serious issues, including latency, huge data and data management. Hence, this work focuses on SaaS, which looks into the handling and managing wisely to maintain the Quality of Service (QoS) and to ensure the users will gain more benefits from the service providers. The people are increasing in number in using the CC for their work including as a service, as a platform or as an infrastructure. The researchers can research and explore more on CC issues due to it significant to people in their daily life.

The increasing the demand usage of those services in brought to latency problem [2] [12] [13]. Therefore, the current works have proposed by using the pre-fetching method even though, by using overaggressive of pre-fetching method it will cause the overhead of the storage [2]. Hence, in this paper, the researcher proposes the use of intelligent method to enhance the current pre-fetching method to improve the CC management. To understand more about intelligent techniques, the details are founds in the next section.

3 Intelligent Method for Cloud Computing

The intelligent method that can apply is the Artificial Intelligence. The intelligent manages machines or software. It also acts as the branch of computer science that develops machines and software with intelligence. The common intelligent techniques that are appropriate for pre-fetching method, which fit for the prediction, including the Artificial Neural Network (ANN) [14] [15], K-Nearest Neighbour (KNN) [14] [15], Decision Tree (DT) [14] [15] and Rough Set (RS) [16]. To see the flow of this research, it can illustrate in taxonomy of intelligent CC as represent in the Fig. 1.

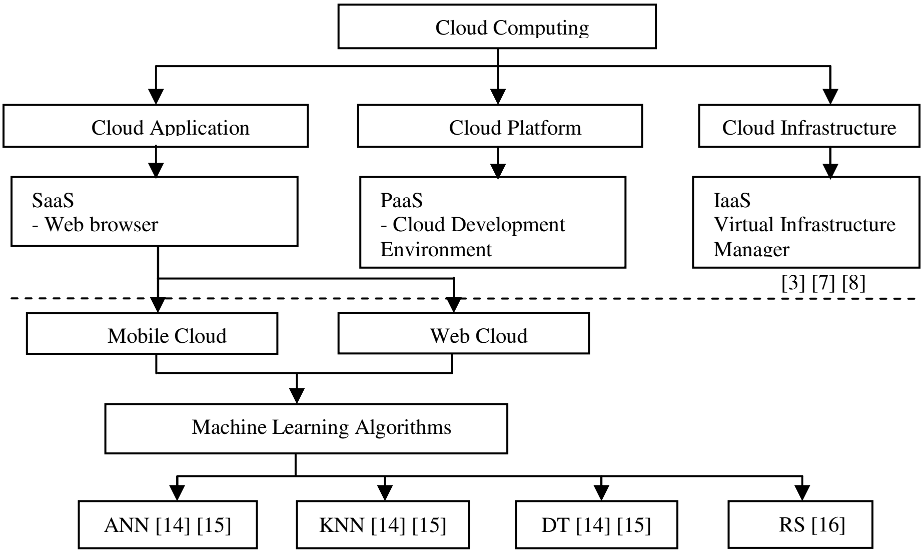


Fig. 1. Taxonomy of Intelligent Cloud Computing

According to Chaudhary et al. [14] it is a great tool in improving the competence and precision of decisions made by intelligent computer programmers. Hence, it is suitable in this work to use the intelligent method because in this work will find the more accurate of prediction data to apply in the real world for reduce the useless data. Besides, it is useful for mobile devices like smart phones, smart cards, and automotive systems [15]. Thus, it is suitable in this research, which looks into mobile-based environment.

3.1 Artificial Neural Network

One of the common intelligent method for pre-fetching is ANN that is approved as a new technology in computer science [17]. ANN acts to achieve high-quality performance during interconnection of simple processing units. A simple conceptual on ANN as depicted in Fig. 2.

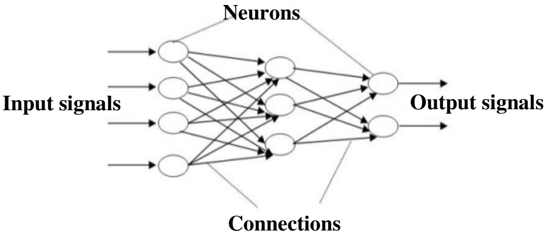


Fig. 2. Artificial Neural Network [17]

ANN has fundamental data about input and output map to be accurate. To train the network a set of data will used. ANN can give any input when the network has been trained and then, it will produce an output, that would communicate to the usual output from the approximated mapping [17], [18]. The capabilities of this ANN, it can provide quality output that is nearly similar to the actual output required [14].

According to Yuan and Yu [19] ANN can be used to enhance the correctness of learning result that is practical with CC. For mutual scenarios, it can proficiently deal with privacy-preserving Back Propagation Network learning, credit to the high scalability of cloud. Besides, it can professionally manage a big data set for learning. However, ANN is a complex effort because in order to reduce over fitting, it needs a great deal of computational effort [15].

Sulaiman et al. [20] used the ANN to train Web caching until it was able to cache the object that the user was more likely to access. Then, to get to the Web caching rules RS was used. The proposed framework proposes to improve the access of social network via mobile devices. In line with Sarwar et al. [21] for high input-output intensive system and implementation of its architecture for better utility in pre-fetching, ANN was used in extracting and recognising the data pattern for case adjustment.

Besides, ANN was the proposed design in cloud, which can validate all types of neural network models [17] in the medical domain. They use SaaS service in cloud to implement the ANN, and the outcomes for the inputs in different layered perceptrons was evaluate in cloud using SaaS and the respective outputs were obtained easily without maintaining any software or any hardware in the system. In their work, the cancer diagnosis was carried out using ANN and the implementation of CC improved the effectiveness and the precision of the diagnosis.

3.2 K-Nearest Neighbor

KNN is one of the most often used classifications, although it can use for estimation and prediction. It basically stores in memory near the training dataset and when a new query is executed, a set of related data are retrieved from the memory that are used for upcoming new data classification [14] [22]. It is referred as KNN because during the classifying, it is useful to consider more than one neighbour at a time [14], [22]–[24].

A study carried out by Chang et al. [25], looked into cloud based intelligent TV programme recommendation system. In order to process user data and generate programme recommendation, the results from the KNN algorithms were be used. It was applied to advise trendy programmes to new users. Nevertheless, due to the limitation on the hardware resource, their study did not use a large number of computers as CC nodes to perform the experiment and analysis. Thus, they could not carry out an experiment to observe the impact of increasing the number of computing nodes on the performance of the proposed system. Hence, they have a chance for applying other algorithms for better performance or precision that can examined in future work.

Moreover, to use the powerful and huge capability of CC, Wang et al. [26] have implemented the KNN that identifies pairs of movies which tended to be rated similarly in order to predict ratings for an unrated item based on ratings of similar items by the same user. The only weakness was that for those whose voting number was less than 100, the number of neighbours for movie-based KNN was somewhat small.

3.3 Decision Tree

DT is describe as a hierarchical model derived from nonparametric theory where local area is recognised in a sequence of recursive splits in a smaller number of steps that implements divide and overcome strategy used in classification and regression tasks. DT is used to analyse a set of pretended training samples; each assigned a class label [27] [28]. The DT system splits the training samples into subsets so that the data in each of the descendant subsets is purer than the data in the parent super set. In order to apply the decision rules for predictions, the input data will be taking. The working flow of a simple DT is shown in Fig. 3.

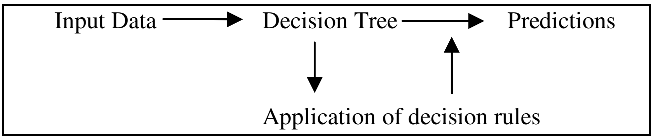


Fig. 3. Decision tree working [14]

The rules can understand by humans and they are use in knowledge system like database. Normally, the algorithms that are commonly used are classification and regression trees (CART) and C4.5. CART, which is suggested by [29], is strictly binary, containing exactly two branches for each decision node. It recursively partitions the records in the training data set into subsets of records with similar values for the target attribute. C4.5 algorithm is an extension from ID3 algorithm for generating DT which recursively visit each decision node, selecting optimal split, until no further splits are possible [30].

According to Liao et al. [27] the effectiveness of different processor for pre-fetch configuration scan greatly influence the performance of memory system and the overall data centre. They have used machine learning like the DT model that can solve parameter optimisation problem. It is a good algorithm and has proper attributes sets, which are feasible to build a precise prediction model for data reduction. Nonetheless, DT represents a complex decision-making system that even a domain expert can hardly compete against.

Next, Nagy et al. [28] have study on educational data mining, whereby a specific data mining field is applied to a set of data that originates from learning environments. They have proposed a framework that uses both classification and clustering techniques to advise recommendations for a certain department for a student or an educational dataset. The experimental result shows the use of C4.5 algorithm that proved to be efficient and robust data. Their work has improved the students' performance and the quality of the education by reducing the failure rate of the first year students.

In addition, Chaudhary et al. [14] have given a review on machine learning technique that can be used for mobile intelligent system. The DT is one of the techniques with powerful and popular implements for classification and prediction. The benefits of using DT learning are it can generate reasonable rules and can perform classification without much computation. Moreover, Shrivastava and Tantuway [16] have proposed DT for reduces the complexity of tree and in addition increases the accuracy. Besides, it provides clear suggestion of which fields are most significant for prediction or classification. However, the limitation of using DT is that it is not appropriate for prediction of continuous attribute and performs badly if used in many classes and small data. Moreover, the computationally is expensive to train the data [14]. Therefore, the research proposes to hybrid the DT technique with another technique to recover and reduce the limitation of DT including it repeat the same attribute that is waste data and select the data without classify properly that know as noise. Thus, the next section covers another technique, which is RS for select the proper data on the compressed data, reduce the complexity and reduce the irrelevant data due to increase the accuracy.

3.4 Rough Set

RS was introduced by Pawlak [31] to overcome the issues with ambiguity and suspicion. Typically, a set of objects is use to analyse. The data that contained the object can presented by a structure called information system. An information system is shown in the information table consists of rows and columns subsequent to objects and attributes. The main advantages of RS are that there is no need for any initial or extra information on data [32]. Then, this technique is easy to manage mathematically. Besides, it uses the simplest algorithms. The dimensionality of the data set is reducing meaningfully by RS theory, which reduces the capacity of storage and makes quicker processing for the algorithm.

As reported by Shrivastava and Tantuway [16], the RS algorithm does not only reduces dimensionality of the dataset, but also provides the best outcome compared to the ID3 algorithm. As stated by Sulaiman et al. [33] they have implemented the RS for optimising mobile web caching performance. From their result, the RS framework for log dataset illustrates to be mutual with an analysis of reduced and derived rules entrenchment of their embedded properties for improved classification outcomes. Along with Chimphlee et al. [34] using RS for web access prediction in order to enhance the prediction measure, which searches the similarity that is determined to work out the resemblance between two sequences. They also planned to use RS for pre-fetching to extract the sequence rules for their future work.

On the other hand, in 2012, Sulaiman et al. [35] provided a guidance for WC concerning of selection for the better parameters to be cached and used in mobile Web pre-caching. In the proxy cache, RS was implemented to enhance the performance of decision Web object was used to either cache or not to cache. This work was extending by merging the per-fetching with the CC and extending the mobile database summarisation, which is the process of reduction of size and information capacity of database. Hence, this work proposes to hybrid the DT with the RS to reduce the complexity of tree and in addition, to increase its accuracy. RS and DT are proposed to hybrid because both are generated by rules for data reduction, which leads to minimal selection of attributes that is the goal of this research; for data reduction [16] [33] [36]. Therefore, it is used to reduce the size of rules to be stored in mobile devices in the future. Table 1 shows the summary of intelligent technique to optimise the pre-fetching method that proposed by other researchers also including the benefit of their research work and the limitation on their proposed work.

Table 1. Intelligent technique for prediction of accuracy

Author(s)	Intelligent Technique	Issues	Benefit	Limitation
Yuan and Yu [19]	ANN	Accurateness of learning result practical with CC	Capable in handling big data set for learning	Complex effort to reduce over fitting and it needs a great deal of computational effort
Sulaiman et al. [20]	ANN	Speed performance for mobile devices	To improve the access of social network using mobile devices	Limited in social network
Sarwar et al. [21]	ANN	Avoiding page faults	Better utility in pre-fetching	For high I/O intensive system only
Rajkumar et al. [17]	ANN	Efficiency and accuracy of diagnosis	Improves the effectiveness and precision of diagnosis	Research in medical domain alone
Chang et al. [25]	KNN	Performance or accuracy of TV recommendation	Improves the TV programme recommendation system	Limitation in observing the result of the rising number of computing nodes
Wang et al. [26]	KNN	Data mining applications and machine learning problem over cloud computing	Enhances the prediction rating	Weakness in small data
Liao et al. [27]	DT	Parameter optimisation problem	Feasible to build a precise prediction model for data reduction	Complex decision-making systems that is hard to compete.
Nagy et al. [28]	DT (C4.5 algorithm)	Accurateness and efficiency of result prediction	Enhances the students' performance and the quality of the education	Scope in education environment alone
Sulaiman et al. [35]	RS	Mobile web caching performance	Reduces and derives rules for enhanced classification result.	Scope in social network alone
Chimphlee et al. [34]	RS	Web access prediction	Enhances the prediction measure	Researches in web domain alone

4 Conclusion

Lot of research is going on in web pre-fetching in various directions. Our research is focus on pre-fetching techniques for CC environment by proposing the intelligent method in enhancing the prediction rule. Therefore, this paper proposes the use of intelligent method that can select a splitting attribute on compressed data, reduce the complexity of tree, increase the accuracy, and reduce unrelated data. From the research, RS is defines to be important with the aim to reduce storage capacity and make the processing of algorithm to faster. Hence, this research proposes the use of this technique via hybrid of the DT and the RS algorithm in order to optimise the performance on mobile web pre-fetching for CC by reducing irrelevant information from the decision table by deleting duplication instance processing of DT. As a result, the MCC system is expect to become faster and decreases the storage size of dataset. The future work will contribute on the testing data set and apply the hybrid technique on CC and will setup the technique for CC environment.

Acknowledgements. This work is supported by Ministry of Education Malaysia (MOE) and Research Management Centre (RMC), Universiti Teknologi Malaysia (UTM). This paper financially supported by Research University Grant (RUG) Tier 1 Program (vote: Q.J130000.2528.03H00). Authors would like to thank Soft Computing Research Group (SCRG), Faculty of Computing (FC), UTM for their continuous support and devotion in making this study a success.

References

1. Prasath, V., Bharathan, N., Lakshmi, N., Nathiya, M.: Fuzzy Logic In Cloud Computing. Int. J. Eng. Res. Technol. 2(3), 1–5 (2013)
2. Ali, W., Shamsuddin, S.M., Ismail, A.S.: A Survey of Web Caching and Prefetching. Int. J. Adv. Soft Comput. Appl. 3(1) (2011)
3. Padhy, P.C., Mishra, S.K.: Cloud Computing: Advance Technique for Corporate Excellence. Int. J. Mech. Eng. Comput. Appl. 1(1), 17–21 (2013)
4. Huth, A., Cebula, J.: The Basics of Cloud Computing, pp. 1–4 (2011)
5. Xing, Y., Zhan, Y.: Virtualization and Cloud Computing. In: Zhang, Y. (ed.) Future Computing, Communication, Control and Management. LNEE, vol. 143, pp. 305–312. Springer, Heidelberg (2012)
6. Fernando, N., Loke, S.W., Rahayu, W.: Mobile cloud computing: A survey. Futur. Gener. Comput. Syst. 29(1), 84–106 (2013)
7. Esseradi, S., Badir, H., Abderrahmane, S., Rattrout, A.: Mobile Cloud Computing: Current Development and Research Challenges. In: 6th Int. Conf. Inf. Technol., ICIT 2013, pp. 1–9 (2013)
8. Neela, K.L., Kavitha, V.: A Survey on Security Issues and Vulnerabilities on Cloud Computing. Int. J. Comput. Sci. Eng. Technol. 4(7), 855–860 (2013)
9. Lauchlan, S.: SaaS at the tipping point as the Big Data Cloud takes shape (2012)
10. Nassif, A.B., Lutfiyya, H.: Measuring the Usage of Saas Applications Based on Utilized Features. In: Proc. 1st Int. Conf. Cloud Comput. Serv. Sci., pp. 452–459 (2011)
11. Gens, F.: Top 10 Predictions IDC Predictions 2013: Competing on the 3rd Platform, IDC, vol. 1, pp. 1–22 (November 2012)
12. Kumar, S.: A Survey on Web Page Prediction and Prefetching Models. Int. J. Comput. Trends Technol. 4(10), 3407–3411 (2013)
13. Gawade, S., Gupta, H.: Review of Algorithms for Web Pre-fetching and Caching. Int. J. Adv. Res. Comput. Commun. Eng. 1(2), 62–65 (2012)
14. Chaudhary, A., Kolhe, S., Kamal, R.: Machine learning techniques for Mobile Intelligent Systems: A study. In: Ninth Int. Conf. Wirel. Opt. Commun. Networks, pp. 1–5 (2012)
15. Larose, D.T.: Discovering Knowledge in Data: An Introduction to Data Mining (2005)
16. Shrivastava, S.K., Tantuway, M.: A Decision Tree Algorithm based on Rough Set Theory after Dimensionality Reduction. Int. J. Comput. Appl. (0975 – 8887) 17(7), 29–34 (2011)

17. Rajkumar, B., Gopikiran, T., Satyanarayana, S.: Neural Network Design in Cloud Computing, Int. J. Comput. Trends Technol. 4(2), 63–67 (2013)
18. Singh, Y., Bhatia, P.K., Sangwan, O.: A review of studies on machine learning techniques. Int. J. Comput. Sci. Secur. 1(1), 70–84 (2007)
19. Yuan, J., Yu, S.: Privacy Preserving Back-Propagation Neural Network Learning Made Practical with Cloud Computing. IEEE Trans. Parallel Distrib. Syst. 25(1), 212–221 (2014)
20. Sulaiman, S., Shamsuddin, S.M., Abraham, A.: Rough Neuro-PSO Web caching and XML prefetching for accessing Facebook from mobile environment, Rough Neuro-PSO WebCaching XML Prefetching Access. Faceb. from Mob. Environ. World Congr. Nat. Biol. Inspired Comput., pp. 884–889 (2009)
21. Sarwar, S., Ul-Qayyum, Z., Malik, O.A.: CBR and Neural Networks Based Technique for Predictive Prefetching. In: Sidorov, G., Hernández Aguirre, A., Reyes García, C.A. (eds.) MICAI 2010, Part II. LNCS, vol. 6438, pp. 221–232. Springer, Heidelberg (2010)
22. Chatzimilioudis, G., Zeinalipour-Yazti, D., Lee, W.-C., Dikaiakos, M.D.: Continuous All k-Nearest-Neighbor Querying in Smartphone Networks. In: IEEE 13th Int. Conf. Mob. Data Manag., pp. 79–88 (July 2012)
23. Donohoo, B.K.: Machine Learning Techniques for Energy Optimization in Mobile Embedded Systems (2012)
24. Jang, M., Yoon, M., Chang, J.: A k-Nearest Neighbor Search Algorithm for Enhancing Data Privacy in Outsourced Spatial Databases 7(3), 239–248 (2013)
25. Chang, J.-H., Lai, C.-F., Wang, M.-S., Wu, T.-Y.: A cloud-based intelligent TV program recommendation system. Computer Electrical Enginering, 1–21 (2013)
26. Wang, J., Wan, J., Liu, Z., Wang, P.: Data Mining of Mass Storage based on Cloud Computing. In: 9th Int. Conf. Grid Coop. Comput. (GCC), pp. 426–431 (2010)
27. Liao, S., Hung, T.-H., Nguyen, D., Chou, C., Tu, C., Zhou, H.: Machine learning-based prefetch optimization for data center applications. In: Proc. Conf. High Perform. Comput. Networking, Storage Anal., Portland, Oregon, pp. 1–10 (2009)
28. Nagy, H.M., Aly, W.M., Hegazy, O.F.: An Educational Data Mining System for Advising Higher Education Students. World Acad. Sci. Eng. Technol. Int. J. Inf. Sci. Eng. 7(10), 175–179 (2013)
29. Breiman, L.E.O.: Random Forests. Mach. Learn. 45(1), 5–32 (2001)
30. Quinlan, J.R.: Learning With Continuous Classes. In: Procceddings AI 1992 Singapore World Sci., vol. 92, pp. 343–348 (1992)
31. Pawlak, Z.: Rough sets. Int. J. Comput. Inf. Sci. 11(5), 341–356 (1982)
32. Triantaphyllou, E., Felici, G.: Data Mining and Knowledge Discovery Approaches Based on Rule Induction Techniques, pp. 1–789 (2006)
33. Sulaiman, S., Shamsuddin, S.M., Abraham, A.: An Implementation of Rough Set in Optimizing Mobile Web Caching Performance. In: Tenth Int. Conf. Comput. Model Simulation, pp. 655–660 (2008)
34. Chimphlee, S., Salim, N., Salihin, M., Ngadiman, B., Chimphlee, W., Srinoy, S.: Rough Sets Clustering and Markov model for Web Access Prediction. In: Proc. Postgrad. Annu. Res. Semin., pp. 470–475 (2006)
35. Sulaiman, S., Shamsuddin, S.M., Abraham, A.: Meaningless to Meaningful Web Log Data for Generation of Web Pre-caching Decision Rules Using Rough Set. In: 4th Conf. Data Min. Optim., vol. 1, pp. 2–4 (2012)
36. Sulaiman, S., Shamsuddin, S.M., Abraham, A.: Rough Set Granularity in Mobile Web Pre-Caching. In: 8th International Conference on Intelligent Systems Design and Applications, vol. 1, pp. 587–592 (2008)

Enhanced Rules Application Order Approach to Stem Reduplication Words in Malay Texts

M.N. Kassim, Mohd Aizaini Maarof, and Anazida Zainal

Faculty of Computing, Universiti Teknologi Malaysia
81310 Skudai Johore, Malaysia
mnizam.kassim@gmail.com, (aizaini,anazida)@utm.my

Abstract. Word stemming algorithm is a natural language morphogical process of reducing derived words to their respective root words. Due to the importance of word stemming algorithm, many Malay word stemming algorithms have been developed in the past years. However, previous researchers only focused on improving affixation word stemming with various stemming approaches. There is no reduplication word stemming has been developed for Malay language thus far. In Malay language, affixation and reduplication are derived words in which have their own morphological rules. Therefore, the use of affixation word stemming to stem reduplication words is considered inappropriate. Hence this paper presents the proposed reduplication word stemming algorithm to stem full, rhythmic and partial reduplication words to their respective root words. This proposed stemming algorithm uses Rules Application Order with Stemming Errors Reducer to stem these reduplication words. Malay online newspaper articles have been used to evaluate this proposed stemming algorithm. The experimental results showed that the proposed stemming algorithm able to stem full, rhythmic, affixed and partial reduplication with better stemming accuracy. Hence, the future improvement of Malay word stemming algorithm should include affixation and reduplication word stemming.

Keywords: word stemming, affixation word stemming, reduplication word stemming, Malay word stemming, Rules Application Order.

1 Introduction

Due to overabundance data on the Internet, there are many research in information retrieval have been conducted for past years. Information retrieval is an area of research focuses on document indexing and retrieval. Information retrieval applications include improved recall and precision of text search, text summarization, machine translation and text categorization. One of important elements in information retrieval is word stemming algorithm [2]. Word stemming algorithm is applied in information retrieval to reduce the size of the text documents for indexing purposes, match the queries with relevant text documents and increase time in retrieving text documents [1] [2] [11]. On the other application, word stemming algorithm is also

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_62, © Springer International Publishing Switzerland 2014

applied in text categorization to reduce high dimensionality in text documents [14]. Regardless of the various applications of word stemming algorithm, the purpose of word stemming algorithm is to reduce the derived words to their respective root word based on morphology structures of natural language [3] [4] [5] [9]. Thus, there have been active research to develop word stemming algorithm for Malay language in the past years. Unfortunately, most of the existing Malay word stemming algorithms only focused on affixation word stemming and gave minimal attention to reduplication word stemming. Both affixation and reduplication words are derived words in Malay language that need to be stemmed in word stemming algorithm. Moreover, the existing Malay word stemming algorithms stem affixation and reduplication words using affixation word stemming in which are considered inappropriate. It is due to affixation and reduplication words have different word morphological structure in Malay language [5][9]. Hence, this paper will propose reduplication word stemming algorithm to stem reduplication words. This paper is organized into four subsequent sections. Section 2 discusses seven word formations in Malay morphology. Section 3 discusses related works on the existing Malay word stemming algorithms and proposed reduplication word stemming algorithm. Section 4 discusses the experimental results and discussion where Malay online news articles have been used to evaluate the proposed stemming algorithm. Finally, Section 5 concludes this paper with a summary.

2 Word Formation in Malay Morphology

Malay language is an Austronesian language which is spoken in Southeast Asia mainly Malaysia, Indonesia, Singapore and Brunei [12]. Unlike other natural languages, Malay language has very complex word morphology where the understanding on how the derived words have been evolved from its respective root words. The word formations in Malay language [5][9][12] are derived from two morphological processes: affixation and reduplication. The next subsections will discussed further these word formations.

2.1 Affixation

In Malay morphology, there are four possible combination between root words and affixes (prefixes, suffixes, confixes and infixes) that that lead to four different categories of affixation: prefixation, suffixation, confixation and infixation. These affixation can be described as follows:

i. prefixation is the combination of root word and prefixes (*ber+, bel+, pe+*) e.g. *berehat* (resting), *belajar* (learning) and *pesakit* (patient)
ii. suffixation is the combination of root word and suffixes *(+an, +i, +kan*) e.g. *makanan* (foods), *hargai* (appreciate) and *kurangkan* (to reduce)

iii. confixation is the combination of root word and confixes (*per+an, me+kan, memper+kan*) e.g. *pertanian* (agriculture), *menyanyikan* (to sing), *mempersiapkan* (to prepare)

iv. infixation is the combination of root word and infixes (+el+, +er+, +em+) e.g. *telunjuk* (pointing), *gerigi* (grill), *gemuruh* (nervous)

In some instances, these affixation words may combine with proclitic (at the beginning of the word e.g. ku+, kau+), enclitic (at the ending of the word e.g. +nya, +mu, +ku) and particles (at the ending of the word e.g. +lah, +kah). For instance, proclitic + affixation word: *kunantikan* (I'm waiting), affixation word + enclitic: *makanannya* (his/her foods) and affixation word + particles: *bergembiralah* (have fun).

2.2 Reduplication

Reduplication is another form of derived words that reflects the plural form of root words in Malay language. There are two different categories of reduplication words: reduplication words with hyphen (full, rhythmic and affixed reduplication) and reduplication without hyphen (partial reduplication). In some instances, there are also

Table 1. Reduplication Word Formation in Malay Morphology

Reduplication Word Category	Reduplication Word	Root Word
Full Reduplication	*gula-gula* (sweets)	*gula*
Rhythmic Reduplication	*rempah-ratus* (spices)	*rempah*
Affixed Reduplication - Prefix I	*berlari-lari* (running)	*lari*
Affixed Reduplication - Prefix II	*berwarna-warni* (colourful)	*warna*
Affixed Reduplication - Prefix III	*surat-menyurat* (letters)	*surat*
Affixed Reduplication - Prefix IV	*pemain-pemain* (players)	*main*
Affixed Reduplication - Suffix I	*buah-buahan* (fruits)	*buah*
Affixed Reduplication - Suffix II	*makanan-makanan* (foods)	*makan*
Affixed Reduplication - Confix I	*keanak-anakan* (childish)	*anak*
Affixed Reduplication - Confix II	*pelajaran-pelajaran* (lessons)	*ajar*
Affixed Reduplication - Confix III	*nasihat-menasihati (*advice)	*nasihat*
Affixed Reduplication - Infix	*tulang-temulang* (bones)	*tulang*
Partial Reduplication	*cecair* (liquid)	*cair*

prefixes, suffixes, confixes and infixes attached to reduplication words. Hence there are 13 possible reduplication word formations in Malay morphology as shown in Table 1.

Other than affixation and reduplication, there are also other word formations in Malay morphology. These words are compounding [e.g. *ambil* (to take) and *alih* (to move)→ *ambilalih* (takeover)] blending [e.g. *cerita pendek* (short story) → *cerpen* (short story)], clipping [e.g. *emak* (mother) → *mak* (mother)], abbreviation [e.g. *orang* (people) → *org* (people)] and borrowing [e.g. *computer* (English) → *komputer* (Malay)]. However, there are possibilities that affixes (*prefixes, suffixes, confixes)* are attached these words to form affixation such as *mengambilalih* (to takeover) and *pengkomputeran* (computing). In short, affixation and reduplication are derived words that need to be stemmed whereas compounding, blending, clipping, acronyms and borrowing are considered as root words that are not to be stemmed. However, the existing Malay word stemming algorithms only focuses on affixation word stemming [1][2][3][4][6][7][8][10][11][13][14] and only two researchers gave minimal attention to reduplication word stemming [4][10]. As shown in Table 1, previous researchers have only considered limited reduplication words: full reduplication rhythmic reduplication and affixed reduplication (Affixed Reduplication - Prefix I, Affixed Reduplication - Suffix I and Affixed Reduplication - Confix I) using affixation word stemming [4][10]. The rest of affixed reduplication and partial reduplication words were not considered in the existing Malay word stemming algorithms.

3 The Proposed Reduplication Word Stemming

After Othman developed the first Malay word stemming algorithm [8] for information retrieval, many researchers have developed subsequent word stemming algorithms for Malay language with various affix removal stemming approaches. These stemming approaches are as follows: original rule-based word stemming [8], modified rule-based - rule application order [2][6][11][13], modified rule application order - rule frequency order [1][3][4], modified rule frequency order [7], modified Porter Stemmer [10] and other method - Boolean extraction [14]. The rules application order was selected to develop the proposed reduplication word stemming algorithm compared to other stemming approaches due to it allows to consider reduplication word stemming and pattern matching rules for all possible reduplication word formations. These rules have been developed using Perl programming v5.5 with regular expression. There are four different categories of reduplication word stemming rules and one category of pattern matching rules in the proposed reduplication word stemming algorithm as illustrated in Figure 1. These reduplication word stemming and pattern matching rules can be described as follows:

```
Input :Accept text document
     Remove HTML tags, special characters except hyphen

Step-1:i = word1,word2,word3,....wordn
       if i = 0, go to Output
      if i = wordn, identify word pattern and segregate
         to respective rules and go to Step-2

Step-2:wordn = specific word pattern
       if wordn = full, rhythmic, affixed reduplication
       Condition 1: check against derivative dictionary
           If wordn found, stem wordn and go to Step-1
           else go to next condition
       Condition 2: stem wordn using full, rhythmic and
         affixed reduplication stemming rules and go to
           Step-1
       if wordn = partial reduplication
           check against lookup dictionary, stem wordn
           and go to Step-1
       If wordn = other words with hyphen
         process wordn using pattern matching rules and
           go to Step-1
     if wordn = other words, drop word and go to Step-1

Output :Root Words{stem1,stem2,stem3,....stemn}
```

Fig. 1. Proposed Reduplication Word Stemming Algorithm

i) There is only single full reduplication word stemming rule due to the nature of full reduplication words that have identical words before and after hyphen such as *burung-burung* (birds), *rumah-rumah* (houses) and *sebab-sebab* (reasons). This reduplication word stemming rule will remove hyphen and word after hyphen e.g. *burung-burung* (birds) → *burung* (bird).

ii) There are 118 rhythmic reduplication word stemming rules due to the nature of rhythmic reduplication words that have words before and after hyphen almost identical but having rhythmic syllable such as *bukit-bukau* (hills), *gunung-ganang* (mountains) and *warna-warni* (colours). These reduplication word stemming rules will remove hyphen and word after hyphen e.g. *bukit-bukau* (hills) → *bukit* (hill).

iii) There are 53 affixed reduplication word stemming rules to due to the nature of affixed reduplication words that have affixation words before and/or after hyphen such as *berlari-lari* (running), *keanak-anakan* (childish), *buah-buahan* (fruits) and *surat-menyurat* (correspondents). These reduplication word stemming rules will remove affixes, hyphen and word after hyphen e.g. *berlari-lari* (running) → *lari* (run). Not all affixes exist in the reduplication words compared to affixation words. Therefore the number of affixed reduplication word stemming rules is less that affixation word stemming rules.

iv) There is only single partial reduplication stemming rule. Partial reduplication words have different word formation from full, rhythmic and affixed reduplication words due to there is no hyphen such as *cecair* (liquid), *dedaun* (leaves) and *lelaki* (men). Thus the partial reduplication word stemming rule contains only single rule to stem partial reduplication by matching the entries in the lookup dictionary that comprises of 39 partial reduplication words and their corresponding root words.

v) There are 21 pattern matching rules to identify other words with hyphen and to drop the hyphen and any remove affixes in the word such as *pro-kerajaan* (pro-government) → *raja* (king), *Kelantan-Kedah* → Kelantan Kedah and *RM450-RM500* → RM450 RM500.

Moreover, the proposed reduplication word stemming algorithm consist of two dictionaries called Stemming Errors Reducers: derivatives dictionary that contains 314 full, rhythmic and affixed reduplications and their respective root words and lookup dictionary that contains 39 partial reduplication words and their respective root words in order to suppress reduplication stemming errors as shown in Table 2.

Table 2. Reduplication Word Stemming with/without Stemming Errors Reducer

Reduplication Words	**Reduplication word stemming without *Stemming Errors Reducer***	**Reduplication word stemming with *Stemming Errors Reducer***
beruang-beruang (bears)	*ruang* (incorrect)	*beruang* (correct)
taman-taman (gardens)	*tam* (incorrect)	*taman* (correct)
perempuan-perempuan (women)_	*rempu* (incorrect)	*perempuan* (correct)
beribu-ribu (thousands)	*ibu* (incorrect)	*ribu* (correct)
penyanyi-penyanyi (singers)	*sanyi* (incorrect)	*nyanyi* (correct)
perompak-perompak (robbers)	*ompak* (incorrect)	*rompak* (correct)

4 Experimental Results and Discussion

The proposed reduplication word stemming algorithm has been evaluated using 270 Malay online articles as testing datasets that consist of 58,563 word occurrences or 8,937 unique words. Out of total unique words, there are 203 full, rhythmic and affixed reduplication words [Dataset A], 5 partial reduplication words [Dataset B], and 97 other words with hyphen [Dataset C]. The first experiment demonstrates that the performance evaluation of full, rhythmic and affixed reduplication word stemming rules against reduplication words with hyphen (Dataset A) have achieved 90.64%

stemming accuracy [e.g. *berlari-lari* (running) → *lari* (run)]. There are three main factors that contributed to stemming errors in the full, rhythmic and affixed reduplication word stemming rules against reduplication words with hyphen: misspelled words, English words and other forms where usually occur in order to continue the word spelling at the end of lines. Then, the second experiment indicates that the performance evaluation of partial reduplication word stemming rules against reduplication words without hyphen (Dataset B) has achieved 100% stemming accuracy [e.g. *cecair* (liquids) → *cair* (melt)]. Interestingly, the result also indicates that there is no stemming errors in partial reduplication word stemming rule against reduplication words without hyphen due to the lookup dictionary is used to stem partial reduplication by matching the entries in the lookup dictionary that comprises of 39 partial reduplication words and their corresponding root words. On other hand, the third experiment indicates that the performance evaluation of pattern matching rules against other words with hyphen (Dataset C) has achieved 88.65% stemming accuracy [e.g. *pro-pembangkang* (pro-opposition) → *bangkang* (oppose), *1960-an* → *1960* and *Kelantan-Perak* → *Kelantan Perak*]. There are four main factors that contributed to stemming errors in the pattern matching rules for other words with hyphen: dates, telephone numbers, loaned prefixes, abbreviations and English words. These other words with hyphen are not considered during the development of pattern matching rules for other words with hyphen. All reduplication stemming and pattern matching rules have been developed based on Malay morphology [5][9]. These experimental results are shown in Table 3.

Table 3. Experimental Results of The Proposed Reduplication Word Stemming

Experiment	Stemming Accuracy	Stemming Errors Samples
Full, Rhythmic and Affixed Reduplication Stemming Rules Against Dataset A	90.64%	▪ misspelled words (*kir-kira, pengusha-pengusaha*) ▪ English words (*re-imaging, co-chairperson*) other forms (*ke-rajaan, memba-ngun)*
Partial Reduplication Stemming Rule Against Dataset B	100%	▪ there is no stemming errors due to the lookup dictionary is used to stem partial reduplication words
Pattern matching rules against other words with hyphen Against Dataset C	88.65%	▪ names (yi-shun → yi, wei-chih → wei) ▪ date (01-aug-2013 → 01) ▪ telephone Numbers (1300-88-5454 → 1300 88) ▪ loaned prefixes (sub-lot → sub lot, wal-jamaah → wal jamaah)

5 Conclusion

In this paper, reduplication word stemming has been proposed. This proposed method uses Rules Application Order (RAO) with Stemming Errors Reducer to stem full, rhythmic, affixed and partial reduplication words. Based on the experimental results, it can be concluded that the proposed reduplication word stemming algorithm produces better stemming results to stem full, rhythmic, affixes and partial reduplication words. Therefore the use of affixation word stemming against reduplication words is considered inappropriate due to the nature of reduplication words with hyphen (full, rhythmic and affixed reduplication), reduplication words without hyphen (partial reduplication) and other words with hyphen (other words) is not similar to affixation words. Hence, the proposed reduplication word stemming algorithm improves reduplication word stemming compared to the existing Malay word stemming algorithms where affixation word stemming is used to stem reduplication words.

Acknowledgments. This research work has been funded by Ministry of Science, Technology and Innovation (MOSTI) Malaysia through Science Fund (Grant No. 01-01-06-SF1167).

References

1. Abdullah, M.T., Ahmad, F., Mahmod, R., Sembok, T.M.T.: Rules frequency order stemmer for Malay language. IJCSNS International Journal of Computer Science and Network Security 9(2), 433–438 (2009)
2. Ahmad, F., Yusoff, M., Sembok, T.M.: Experiments with a Stemming Algorithm for Malay Words. Journal of the American Society for Information Science 47(12), 909–918 (1996)
3. Darwis, S.A., Abdullah, R., Idris, N.: Exhaustive Affix Stripping And A Malay Word Register To Solve Stemming Errors And Ambiguity Proble. In: Malay Stemmers. Malaysian Journal of Computer Science (2012)
4. Fadzli, S.A., Norsalehen, A.K., Syarilla, I.A., Hasni, H., Dhalila, M.S.S.: Simple Rules Malay Stemmer. In: The International Conference on Informatics and Applications (ICIA 2012), pp. 28–35. The Society of Digital Information and Wireless Communication (2012)
5. Hassan, A.: Morfologi, vol. 13. PTS Professional (2006)
6. Idris, N., Syed, S.M.F.D.: Stemming for Term Conflation in Malay Texts. In: International Conference on Artificial Intelligence, ICAI 2001 (2001)
7. Leong, L.C., Basri, S., Alfred, R.: Enhancing Malay Stemming Algorithm with Background Knowledge. In: Anthony, P., Ishizuka, M., Lukose, D. (eds.) PRICAI 2012. LNCS, vol. 7458, pp. 753–758. Springer, Heidelberg (2012)
8. Othman, A.: Pengakar Perkataan Melayu untuk Sistem Capaian Dokumen. MSc Thesis. Universiti Kebangsaan Malaysia, Bangi (1993)
9. Ranaivo-Malancon, B.: Computational Analysis of Affixed Words in Malay Language. In: Proceedings of the 8th International Symposium on Malay/Indonesian Linguistics, Penang, Malaysia (2004)

10. Sankupellay, M., Valliappan, S.: Malay Language Stemmer. Sunway Academic Journal 3, 147–153 (2006)
11. Sembok, T.M.T., Yussoff, M., Ahmad, F.: A Malay Stemming Algorithm for Information Retrieval. In: Proceedings of the 4th International Conference and Exhibition on Multi-Lingual Computing, vol. 5, pp. 2–1 (1994)
12. Sharum, M.Y., Abdullah, M.T., Sulaiman, M.N., Murad, M.A., Hamzah, Z.Z.: MALIM - A new computational approach of Malay morphology. In: 2010 International Symposium on Information Technology (ITSim), vol. 2, pp. 837–843 (2010)
13. Tai, S.Y., Ong, C.S., Abdullah, N.A.: On Designing An Automated Malaysian Stemmer For The Malay Language. In: Proceedings of the Fifth International Workshop on Information Retrieval With Asian Languages, pp. 207–208. ACM (2000)
14. Yasukawa, M., Lim, H.T., Yokoo, H.: Stemming Malay Text and Its Application in Automatic Text Categorization. IEICE Transactions on Information and Systems 92(12), 2351–2359 (2009)

Islamic Web Content Filtering and Categorization on Deviant Teaching

Nurfazrina Mohd Zamry, Mohd Aizaini Maarof, and Anazida Zainal

Faculty of Computing, Universiti Teknologi Malaysia, Johor 81310, Malaysia
nurfazrina.mohdzamry@gmail.com, {aizaini,anazida}@utm.my

Abstract. Currently, process for blocking the deviant teaching website is done manually by Malaysia authorities. In addition there are no Web filtering product offered to filter religion content and especially for Malay language. Web filtering can be used as protection against inappropriate and prevention of misuse of the network and hence, it can be used to filter the content of suspicious websites and alleviate the dissemination of such Web page. The purpose of the paper is to filter the deviant teachings Web page and classify them into three categories which are deviate, suspicious and clean. There are three Term Weighting Scheme techniques were used as feature selection included Term Frequency Inverse Document Frequency (TFIDF), Entropy and Modified Entropy. Support Vector Machine (SVM) will be used for classification process. As a result, M. Entropy shows the most suitable term weighting scheme to use in Islamic web pages filtering rather than TFIDF and Entropy.

Keywords: Web Content Filtering, Deviant Teaching, Term Weighting Scheme.

1 Introduction

Nowadays, Internet became faster compare to the last twenty years which offers thousands of information in the cloud. Unfortunately, the information provided sometimes cannot be identified either it was genuine or fake, especially when it involves with critical issues that may bring harm to the society. As mentioned by [1], not all of this online content is accurate, pleasant, or inoffensive. Thus, it became the biggest challenge people face from the Internet to validate the accuracy of content in the Internet.

Among the critical and sensitive issues in the internet are the religious belief, political issue and pornography. In [2], the evolution of electoral politics on the Web have been analysed based on extensive analysis of hundreds of campaign sites produced by candidates in U.S. elections in 2000, 2002, and 2004. In conclusion Web becoming more convenient way to capture the followers. Moreover, there are some of the practitioners of deviant teachings took this opportunity to attract followers just using the Internet especially to distort beliefs of Muslim in Malaysia. Consequently, Muslim which does not have a solid foundation of Islam may simply be attracted to this group.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_63, © Springer International Publishing Switzerland 2014

For solution, Web filtering is suitable application which has provided two major services: protection against inappropriate and prevention of misuse of the network [3]. Since, some of the content may threaten the society as there will reflecting in violence, hate as well as undermine national harmony. Hence, the Web filtering system can be used to filter the content of suspicious websites and alleviate the dissemination of such website. A lot of web filtering can be found nowadays focused on monitoring pornography, parental control and internet security content filtering. As far as our concerned, still there is no web filtering products focused on religion content especially for Muslim community in Malay. Therefore, this paper will be concerned in Web filtering of deviant teaching issue. On the other hand, this research can be used as the foundation to develop Islamic filtering product in the future.

The rest of the paper will be arranged as follows; the situation of deviant teaching in Malaysia is further highlighted in Section 2. Section 3 provides an overview of Web content filtering and existing application of Web content filtering. In Section 4, the details of techniques used for Web content filtering is highlights. The methodology used in this research is discussed in Section 5. Section 6 discussed the result, discussion and suggests further opportunities for future research. Section 7 concludes this survey.

2 Deviant Teaching in Malaysia

As Malaysia is Islamic country, fighting the deviant teaching can be the vital issues for authorities. There are several practices of this deviant teaching identified by the religious authorities in Malaysia. Among this group, *Al-Arqam, Syiah, Ajaran Martabat Tujuh, Ajaran Ayah Pin, Al-Ma`unah*, certain type of *Tarikat* and *Wahdat al-Wujud* have been magnificently renown all over Malaysia. Since Internet provide an easy and fast way to seek information [4], it has been used as a medium to attract the follower. Easy making information on the Web can led some individuals to put up harmful materials on the Web. These give the advantages for practitioners of deviant teachings to disseminate their doctrine. Nevertheless, these may cause some negative impact on beliefs, attitudes and practices of the Muslim community in the country.

Malaysian Department of Islamic Development (JAKIM) is one of the government organizations that responsible for any issues that related to Islamic society. However, Web content filtering is done manually by JAKIM. According to experts from Faculty of Islamic and Civilization of Universiti Teknologi Malaysia, there is no specific Web filtering software that focuses on deviant teaching especially in Malay language. Manual Web filtering processes start when JAKIM receive a report from people either by telephone or via email. Special unit in JAKIM then investigate the reported website to filter the website whether it may contain the deviant teaching issues where the investigation normally last for few months. Since the investigation process takes long period, there is a possibility that some of Internet user may accept the doctrine.

3 Web Pages Filtering

Web filtering software is software which is specifically designed and optimized to controlling what content is permitted to users, especially when it is used to filter material delivered over the Web. Web filtering is commonly used by organizations such as offices and schools to prevent computer users from viewing inappropriate web sites or content, or as a pre-emptive security measure to prevent access of known malware hosts [5].

3.1 Overview of Web Content Filtering System

There are four types of web filtering approaches mentioned by [6] included Platform for Internet Content Selection (PICS), Uniform Resource Locator (URL) blocking, keyword filtering and intelligent content analysis. Intelligent content analysis is the best to classify the web content since it can be used to categorize Web pages into different groups. Meanwhile, keyword filtering sometimes will over-block the Web while URL blocking keeps maintaining the reference list which gives some disadvantages to filter the Web.

Intelligent content analysis was started when there are some limitations on URL filtering keyword based filtering and PICS technique. Text based content analysis approach was the first Intelligent content analysis introduced. Another intelligent content analysis but based on image have been one of the research trend during 2008. Latest experiment was carried out in [4] , on enhancing the Web filtering using three different features selection; term weighting scheme, features extraction method on dimensional reduction algorithm and hybrid of both method.

3.2 Web Content Filtering: Text Classification

Generally, the implementation of Web content filtering with the text classification in information categorization and filtering is one of the techniques that broadly used

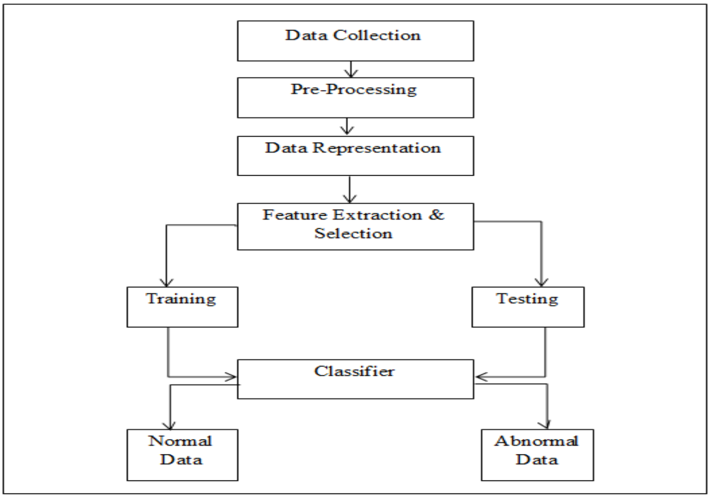


Fig. 1. Architecture of Intelligent Content Analysis [4]

nowadays. Still there are very limited implementations of Web filtering in Malay-based especially in the deviant teaching. Fig. 1 shows a general architecture of Web Intelligent content filtering with text classification.

The most vital part in this architecture is pre-processing process (HTML parsing, Stemming and Stopping). This paper implemented Enhance Sembok's stemming algorithm introduced by [7] which is the enhanced from Sembok's algorithm. Enhance Sembok's stemming algorithm were used to cater stemming Malay term and this algorithm will be explained more in next section as well as the other techniques used for Intelligent Content Analysis.

3.3 Existing Web Content Filtering Application

As far as our concerned there is still no web content filtering application exists in deviant teaching domain especially in Malay language. Most of web content filtering products are designed to filter the pornography or illicit Web pages namely NetNanny, SurfWatch, Safe Eyes, Cyber Sitter and Secure Web Smart Filter which enforced in different level filtering.

4 Techniques Used for Web Content Filtering

This section will discussed all the techniques used in this research to perform Web content filtering on deviant teaching Web pages. The techniques include the Enhance Sembok's stemming algorithm used in Pre-processing phase, Term-Weighting Scheme (TWS) used for Features Selection and SVM as Classifier.

4.1 Stemming

Stemming process used to split the root word from the words using in the Web pages. The removal of suffixes in English and similar languages, Slovene and French, are found to be sufficient for the purpose of information retrieval but this is not so in Malay [8]. This is due to the fact that Malay affixes consist of four different types

Step-1 : Check the word against the dictionary, if word found. Accept it asroot word and exist;otherwise proceed to next step;

Step-2 : Check the word in the added dictionary, if exists, accept as root word and exit; otherwise proceed;

Step-3 : Check the given pattern of the rule with the word; If the system find a match. Apply the rule to the word to get a stem;

Step-4 : Check the stem against the dictionary; if exists, accept as root word; otherwise **proceed**

Step-5 : Apply the Rule Two, if match, accept as root word; if not, rechecking

Fig. 2. Enhance Sembok's Stemming Algorithm [7]

of verbal elements [9] which are prefix, suffix, infix, and prefix-suffix. Lovins and Potter algorithm are example of stemming algorithm used for English language while Othman and Sembok's is used for Malay language. According to [7] two changes made from Sembok's stemming algorithm in step 2and step 5 as illustrated in Fig. 2**.** The enhancement made use to cater the problem of spelling error during the stemming process occurred in Sembok's stemming algorithm.

4.2 Feature Selection Method in Web Filtering

A Term-Weighting Scheme (TWS) is essentially the document ranking function which its assign values to search terms based on how useful they are likely to determine the relevance of a document [10]. Three main factors used in TWS include term frequency, collection frequency and length normalization as stated by [11]. In this paper the Term Frequency Inverse Document Frequency (TFIDF) Entropy and M. Entropy will be used for TWS. More information about the TWS will be discussed in section 5.2.

4.3 Classification Method

In this paper, the classification will be done based on deviate, suspicious and clean Web pages. There are many approaches used to classify the Web pages such as Naïve Bayes, Decision Tree and Artificial Neural Network. Support Vector Machine (SVM) was chosen for classification process for this paper. SVM is one of the supervised classification method that uses training data in the system and configuration that data as a learning model to predict the data category and SVM is also applied to text classification [3]. All these techniques are broadly applied in Web content filtering process especially in pornographic filtering. Unfortunately filtering deviant teaching Web did not get much attention as compared to pornography or illicit Web content filtering in the market because there is no any enforcement or awareness in Muslim society about the needs of deviant teaching Web filtering.

5 Methodologies

The methodology of this study composes of three phases and its follow the architecture of Intelligent Content Analysis as mentioned in section 3.2. The phase includes term identification, feature selection method in web filtering and classification.

5.1 Phase 1: Term Identification

The main objective of first phase is to pre-process and identify the deviant teaching keywords from the Web pages. In data collection stage, the expected outcome is establishing the dictionary and stop list collected from the internet. Pre-processing stage is one of the important process since it will be the fundamental of the whole research where pre-processing will be produced the deviant teaching keyword. To convert the web pages, it involves three processes; HTML parsing, stemming and stopping. Since this research purposely used as the proof of concept, the term are only collected from selected deviant teaching group only.

5.2 Phase 2: Feature Selection Method in Web Filtering

Feature selection is based on the reaction of the cross-validation dataset classification error due to the removal of the individual features and feature selection is a special case of feature extraction [12]. Term Weighting Scheme (TWS) is the standard procedure commonly used in text classification [13]. These TWS assign values to search terms based on how useful they are likely to be in determining the relevance of a document [10]. In this paper the Term Frequency Inverse Document Frequency (TFIDF), Entropy and M. Entropy will be used for term weighting scheme.

TFIDF was introduced by [14] and the equation for TFIDF is given below:

Given N is total number of document collection, i is a word and $j \in N$ then;

$$\mathrm{x}ij = TFij \times IDFi \tag{1}$$

$$TFij = \begin{cases} 1 + \log 10\ \mathrm{TFij} & (\mathrm{TFij}\ \geq 0) \\ 0 & (\mathrm{TFij}\ = 0) \end{cases} \tag{2}$$

$$IDFi = \log 10(NDFi) \tag{3}$$

x_i is the total number term weight of term in collection. TF_{ij} represent the frequent number of ith term appear on jth document while DF_i is the number of document contain ith term in the collection. Entropy weighting scheme was used by [15] for new classification and the equation is given below:

$$\mathrm{Gij} = \frac{1 + \sum_{i=1}^{n} \frac{TFij}{Fi} \log\left(\frac{TFij}{Fi}\right) + 1}{\log N} \tag{4}$$

$$\mathrm{L}ij = \begin{cases} 1 + log\ \ TFij & (\mathrm{TFij}\ > \ 0) \\ 0 & (\mathrm{TFij}\ = \ 0) \end{cases} \tag{5}$$

$$\mathrm{X}i = \mathrm{L}_{\mathrm{ij}}\ \mathrm{x}\ \mathrm{G}_{\mathrm{i}} \tag{6}$$

For Entropy equation, L_i are assigning as local term weight of term, ith in jth document ang G_i as global term weight. The other notation is similar to equation of (2.1),(2.2) and (2,3). Meanwhile, modification is made for that G_i and Li for $lenDoc_j$ denotes as total words exist in j^{th} document where it concern about the length of document for equation of M. Entropy which proposed by [6] as given below:

$$\text{Gij} = \left(\frac{\log 10\, DFi}{\log 10\, N}\right) + 1 \tag{7}$$

$$\text{L}ij = \begin{cases} \left(\frac{log10\,(TFij)}{(\log 10\, lenDocj)} + 1\right) \times \left(\frac{log10\,(TFij)}{log10\, T(i)} + 1\right) & (TFij > 0) \\ 0 & (TFij = 0) \end{cases} \tag{8}$$

$$\text{T}i = \sum_{j=1}^{n} TFij \tag{9}$$

$$\text{W}_{\text{ij}} = \text{L}_{\text{ij}} \text{ x } \text{G}_{\text{i}} \tag{10}$$

5.3 Phase 3: Classification

Approximately about 300 web pages from the Internet were collected to form sample of data and this sample will be divided into three categories which are deviant (Dev), suspicious (Sps) and clean (Cln). The specifications of SVM are summarized in Table 1.

Table 1. Specifications of SVM

Parameter	Description	Value
S	Type of SVM	C-SVC
K	Type of kernel function	Linear: u' * v
D	Degree in kernel function	3
G	Gamma in kernel function	0.001
R	Coefficient in kernel function	1
Z	Whether to normalize input data	0
M	Cache memory size in MB	40
E	Tolerance of termination criterion	0.001
H	Whether to use the shrinking heuristics	1
W	The parameters C of class i to weight[i]*C	1

The dataset have been divided into three set to analyse the accuracy and performance of each category. Each dataset have specific purpose for the experiment. Dataset 1 consists of deviate and suspicious web pages. The purpose of this dataset collection is to select the most relevant features when both dataset share almost similar terminology, but the content is clearly different from each other. Dataset 2 consists of deviate and clean web pages. Since this dataset is very different from each other hence, the purpose of this dataset collection is to select the most relevant features when both datasets are in normal condition. Dataset 3 composes all three categories. This dataset is purposely created to select the most relevant features when datasets are in real environment condition. Table 2 shows the total number of data to process in the experiment.

Table 2. Dataset for Experiment Process

Dataset	Dataset 1			Dataset 2			Dataset 3			
Category	**Dev**	**Sps**	**Total**	**Dev**	**Cln**	**Total**	**Dev**	**Sps**	**Cln**	**Total**
Training	60	60	120	60	60	120	60	60	60	180
Testing	40	40	80	40	40	80	40	40	40	120
Total	100	100	200	100	100	200	100	100	100	300

6 Results and Discussion

Results from the experiment will be highlighted in this section and it follows by some discussion. Firstly, list of top terms with feature ranking before and after expert intervention were shows in Table 3.

Table 3. List of Top Terms with Feature Ranking Before and After Expert Intervention

	TFIDF		Entropy		M.Entropy	
#	**Before**	**After**	**Before**	**After**	**Before**	**After**
1	ajaran	mirza	ajaran	al	al	al
2	wahyu	abuya	islam	muhammad	islam	muhammad
3	mirza	ghulam	al	imam	ajaran	imam
4	abuya	martabat	allah	rasul	allah	manusia
5	ghulam	mahdi	muhammad	ahmad	muhammad	ahmad
6	martabat	nur	ilmu	manusia	imam	suci
7	mahdi	cahaya	imam	akidah	ilmu	akidah
8	nur	ahmad	rasul	suci	rasul	mahdi
9	pemimpin	ahmadiyah	quran	wal	ahli	hakikat
10	cahaya	wal	ahli	mahdi	manusia	wal

Classification process takes 100 features or terms for each dataset and TWS. In this research, features are defined as the term of the deviant teaching. The term need to undergo expert intervention to select only relevant and significant term to get more accurate Web filtering result. Table 4 shows the result of accuracy for all three TWS.

Table 4. Accuracy Result of Term Weighting Scheme

	Dataset 1			Dataset 2			Dataset 3		
#	**TFIID**	**Entropy**	**M.Entropy**	**TFIID**	**Entropy**	**M.Entropy**	**TFIID**	**Entropy**	**M.Entropy**
25	56.25	48.75	95.00	65.00	100.00	51.25	56.25	48.75	95.00
50	50.00	47.50	76.25	58.75	100.00	50.00	50.00	47.50	76.25
75	48.75	45.00	55.00	43.75	100.00	50.00	48.75	45.00	55.00
100	50.00	48.75	50.00	51.25	100.00	95.00	50.00	48.75	50.00

Fig 3 illustrates the classification accuracy result of all three TWS using dataset 1. M. Entropy shows the highest accuracy in features 25 to 75. However, the value of M. Entropy was decreased when the number of features increased. Meanwhile, the graph changes for TFIDF and Entropy were not too obvious. In overall, classification can achieve highest result in the beginning for all term weighting but it will decrease when the features increased. In addition, all TWS with 25 and 100 features always shows the highest result. Unfortunately, M. Entropy shows a big drop values rather than TFIDF and Entropy which seems more consistent. This mean term frequency and document length are the main factors for filter the Web pages. Nevertheless, dataset 1 which contains of deviant and suspicious categories can be the other reason discriminate those Web pages.

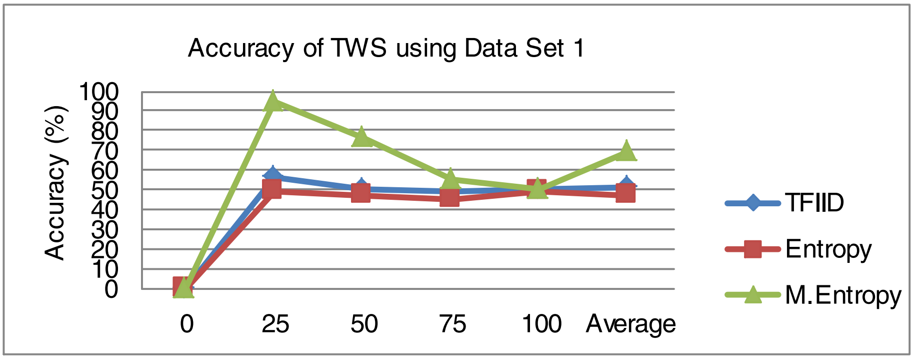


Fig. 3. Accuracy Result of Different TWS for Dataset 1

On the other hand, result obtained in dataset 2 shows Entropy achieved higher result compared to TFIDF and M. Entropy as illustrates in Fig. 4. Similar to dataset 1, TFIDF performance is consistent compared to the other two. As the dataset 2 contains Web pages from deviate and clean categories hence, it's easier to discriminate the Web pages than previous dataset 1. However, M. Entropy shows different graph shapes than dataset 1. The accuracy increased when the numbers of features achieved 100 features but graph drop in the middle. This is because M.Entropy make assumption that shorter the document more strong the term weight. However,

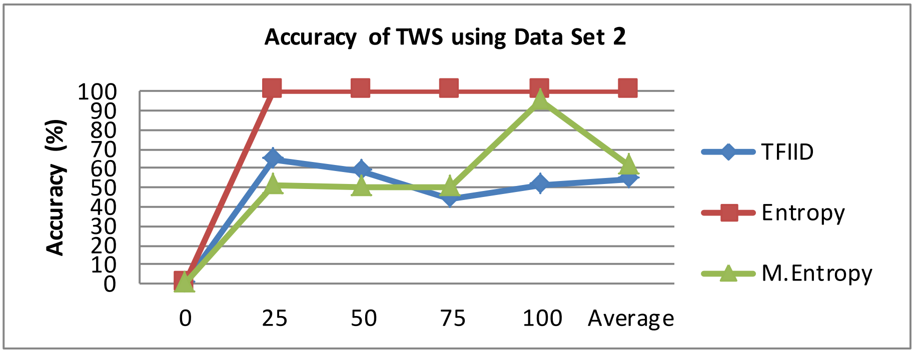


Fig. 4. Accuracy Result of Different TWS for Dataset 2

this assumption was tested for pornographic Web pages. It may not be accurate when evaluates the deviant teaching Web pages. Even M. Entropy shows highest value when tested with 100 features, but the big gap between value of accuracy for 75 features and 100 features make it not feasible to use as TWS.

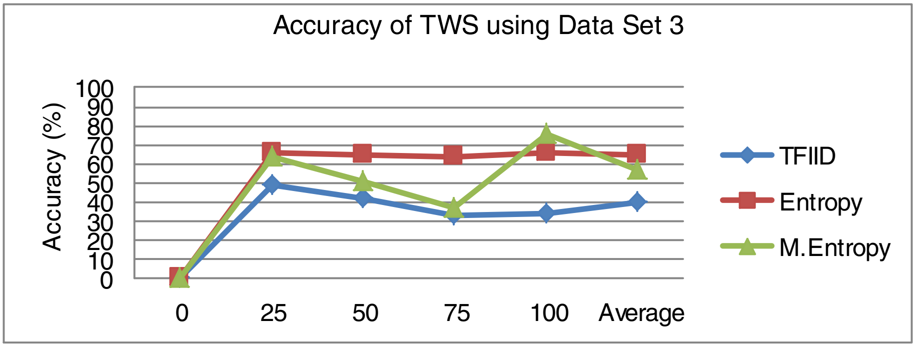


Fig. 5. Accuracy Result of Different TWS for Dataset 3

In Fig 5, TFIDF still acted consistent as previous dataset when graph shows there is not obviously fluctuated. While, Entropy is more consistent as it shows only few changes compare to other two TWS shows in dataset 3. Again Entropy might be suitable TWS in deviant teaching web filtering as it give highest accuracy compared to TFIDF and M. Entropy. As compared to dataset 2, the graph of M. Entropy in dataset 3 shows the same classification performance which; 1) the accuracy increased when it used 100 features, 2) accuracy value was high when for 25 features and 3) value was decreased in the middle. As dataset 3 purposely to examine the ability of TWS when tested with real environment condition hence, the accuracy shows low value rather than dataset 2. Moreover, it suited the assumption made for M.Entropy when 25 features show high accuracy but decreased when features increased. However, this assumption is might not feasible to use for deviant teaching Web pages when value was increased when 100 features are examined.

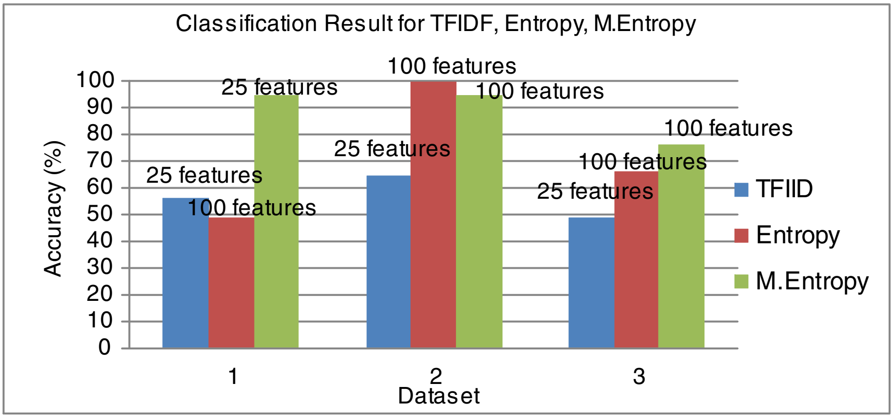


Fig. 6. The Summarization of Classification Result for Dataset 1 to 3

Fig. 6 shows the summary of all dataset (1 – 3). The value is taken based on the highest value achieved from each dataset. The values of highest result may come from different features.

Fig. 6 shows that M. Entropy has achieved highest accuracy in average compared to TFIDF and Entropy. Accept for dataset 1, the highest accuracy achieved when examined 100 features. This means, M. Entropy might be a relevant TWS for filtering the deviant teaching websites. Since Entropy shows a big gap and inconsistency between the accuracy of dataset 1, 2 and 3 this might results on incompatible to filter the web pages. Similar to M. Entropy, Entropy can easily discriminate data in dataset 2 but was not able to discriminate in dataset 1. This could be it is difficult to classify the suspicious and deviant content in dataset 1 rather than deviant and clean content in dataset 2. On other hand, TFIDF always gives the lowest value in all the datasets which can be assume that TFIDF is not suitable term weighting scheme for deviant teaching Web pages. This is because TFIDF was not concerned with the term frequency and document length like M. Entropy. Moreover, highest values were obtained when only 25 features selected.

7 Conclusions

In conclusion, M. Entropy might be the better feature selection method to be used in deviant teaching Web filtering in Malay language. However, Web pages filtering in Malay language seem to be more complex compared to English language since Malay word can have double meaning and converting Arabic term to Malay can have different version of term. In addition, when it comes to deviant teaching, term need to consider in context rather than single term. Moreover, there are lots of deviant teaching groups out there and each group have their specific term portrait their groups. Thus, all this affected the performance for filtering the web pages. Hence, to improve performance this Web content analysis need to be improved especially on pre-processing as well as features selection part.

After the research completed, it might give a solution on providing the proactive web pages filtering system rather than reactive system which web content are filtered manually. Hence it beneficial those the Islamic society and authorities like JAKIM, school and even personal used.

References

[1] Heins, M., Cho, C., Goldberg, D.: Internet Filters (2006)

[2] Foot, K.A., Schneider, S.M.: Web Campaigning, p. 263. MIT Press (2006)

[3] Du, R., Safavi-Naini, R., Susilo, W.: Web filtering using text classification. In: The 11th IEEE International Conference on Networks, pp. 325–330 (October 2003)

[4] Lee, Z.S.: Enhanced Feature Selection Method For Illicit Web Content Filtering. Universiti Teknologi Malaysia (2010)

[5] Salleh, S.F.M.: Comparative Study On Term Weighting Schemes As Feature Selection Method For Malay Illicit Web Content Filtering. Universiti Teknologi Malaysia (2012)

[6] Lee, Z.-S., Maarof, M.A., Selamat, A., Shamsuddin, S.M.: Enhance Term Weighting Algorithm as Feature Selection Technique for Illicit Web Content Classification. In: 2008 Eighth Int. Conf. Intell. Syst. Des. Appl., pp. 145–150 (November 2008)

[7] Mazlam, N.: Enhancement of Stemming Process for Malay Illicit Web Content. Universiti Teknologi Malaysia (2012)

[8] Sembok, T.M.T., Bakar, Z.A., Ahmad, F.: Experiments in Malay Information Retrieval. In: 2011 International Conference on Electrical Engineering and Informatics (July 2011)

[9] Fadzli, S.A., Norsalehen, A.K., Syarilla, I.A., Hasni, H., Satar, S.D.M.: Simple rules malay stemmer. In: The International Conference on Informatics and Applications (ICIA 2012), pp. 28–35 (2012)

[10] Cummins, R., O'Riordan, C.: Evolved term-weighting schemes in Information Retrieval: an analysis of the solution space. Artif. Intell. Rev. 26(1-2), 35–47 (2007)

[11] Yang, Y., Pederson, J.O.: A Comparative Study on Feature Selection inText Categorization. In: Proceedings of the Fourteenth International Conference on Machine Learning, ICML 1997, pp. 412–420 (1997)

[12] Verikas, A., Bacauskiene, M.: Feature selection with neural networks. Pattern Recognit. Lett. 23(11), 1323–1335 (2002)

[13] Liu, Y., Loh, H.T., Sun, A.: Imbalanced text classification: A term weighting approach. Expert Syst. Appl. 36(1), 690–701 (2009)

[14] Salton, G., Wong, A., Yang, C.S.: AVector Space Model for Automatic Indexing. Commun. ACM 18(11) (1975)

[15] Selamat, A., Omatu, S.: Web page feature selection and classification using neural networks. Inf. Sci. (Ny). 158, 69–88 (2004)

Multiobjective Differential Evolutionary Neural Network for Multi Class Pattern Classification

Ashraf Osman Ibrahim[1,2], Siti Mariyam Shamsuddin[1], and Sultan Noman Qasem[3]

[1] Soft Computing Research Group (SCRG), Faculty of Computing,
Universiti Teknologi Malaysia (UTM), 81310, Skudai, Johor, Malaysia
Ashrafosman2@gmail.com
[2] Faculty of Computer and Technology, Alzaiem Alazhari University, Khartoum, Sudan
[3] College of Computer and Information Sciences,
Al-Imam Muhammad Ibn Saud Islamic University, Riyadh, Saudi Arabia

Abstract. In this paper, a Differential Evolution (DE) algorithm for solving multiobjective optimization problems to solve the problem of tuning Artificial Neural Network (ANN) parameters is presented. The multiobjective evolutionary used in this study is a Differential Evolution algorithm while ANN used is Three-Term Backpropagation network (TBP). The proposed algorithm, named (MODETBP) utilizes the advantages of multi objective differential evolution to design the network architecture in order to find an appropriate number of hidden nodes in the hidden layer along with the network error rate. For performance evaluation, indicators, such as accuracy, sensitivity, specificity and 10-fold cross validation are used to evaluate the outcome of the proposed method. The results show that our proposed method is viable in multi class pattern classification problems when compared with TBP Network Based on Elitist Multiobjective Genetic Algorithm (MOGATBP) and some other methods found in literature. In addition, the empirical analysis of the numerical results shows the efficiency of the proposed algorithm.

Keywords: Multiobjective Differential Evolution, Three-Term Back-propagation network, Pareto optimal, classification.

1 Introduction

Over the past few decades, there was a significant increase in using soft computing approaches. Artificial Neural Network (ANN) has become the substrate of soft computing methods, successfully used for solving different problems. Due to the importance of using ANNs in many applications, there are some different methods in the previous studies that focused on solving the problems of ANNs optimization, the training and structure of the network [1, 2].

Recently, there has been a remarkable increase in the use of Evolutionary algorithms (EAs) for solving optimization problems. The design of ANNs is considered one of the most important problems that need to be solved using this kind of algorithms. The earlier approaches tackled the single objective optimization

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_64, © Springer International Publishing Switzerland 2014

problems in some of the previous works, PSO [3], GA[2] and DE[4] were considered for optimizing ANNs. These optimization techniques optimize only one factor, such as, hidden nodes or connection weights or optimizing training error rate. Though in ANNs optimization, there is more than one parameter that need to be optimized. Therefore, multiobjective optimization problems are preferred because of their ability to optimize more than one objective simultaneously.

Evolutionary Algorithms (EAs) are good candidates for Multi objective optimization problems (MOOPs). This is because of their abilities to search for multiple Pareto optimal solutions and they perform better in global search space. Multiobjective evolutionary algorithms (MOEAs) research area has become one of the hottest areas in the field of evolutionary computation [5]. They are suitable to produce and design the appropriate and accurate ANNs with the optimization of two conflicting objectives, namely: minimization of ANNs structure complexity and maximization of network capacity. Hence, the MOEAs have been successfully applied recently to optimize both the structure, connection weights and network training simultaneously [6-8]. These methods have advantages over the conventional backpropagation (BP) method because of their low computational requirement when searching in a large solution space due to the fact that EAs are population based algorithms which allow for simultaneous exploration of different parts in the Pareto optimal set. Thus, Pareto optimal solutions are used to evolve artificial neural networks (ANNs) which are optimal both with respect to classification accuracy and network architectural complexity [9, 10].

The MOEAs have been receiving increased attention among researchers to solve this kind of problem. Recently, the trend to optimize and design ANNs architecture by using MOEAs is gaining popularity among researchers. It still attract significant interest of the evolving ANNs as optimization problems. In this paper, we proposed to use a MODE algorithm to optimize and design appropriate and accurate ANNs architecture, that will be able to find an appropriate number of hidden nodes in the hidden layer along with error rates of the network. The proposed method benefited from the concept of the Pareto optimal solutions, using Multiobjective Differential Evolution (MODE) for designing a good structure of ANNs for the multi class classification task.

The other aspects which the paper delves on includes: Related Works given in Section 2. In Section 3, Material and methods are presented. The proposed method is presented in Section 4. In Section 5, Experimental Study, result and discussion are provided. Finally Section 6 presents the conclusions.

2 Related Works

There are various algorithms that have been proposed for ANNs with MOEAs. However, due to a large diversity of applications, various data and different purposes of optimization algorithms, it is so difficult to find a specific algorithm that can serve the needs of all, at the same time. Zhou, A., et al. [5] presented a survey paper on the development of Multiobjective evolutionary algorithms. The survey covers all areas that apply different types of MOEA to real world problems, such as, data mining, communications, bioinformatics, control systems and robotics, manufacturing, engineering, pattern recognition, image processing, fuzzy systems and Artificial neural networks. Furthermore, they presented some issues for the future work. In addition, they highlighted that the MOEAs are very promising for multiobjective optimization problems because of their capability to approximate a set of Pareto optimal solutions in a single run.

Another study has proposed a solution for the regularization problems of complexity in the networks, based on multi-objective optimization algorithm to optimize the structure and minimize the number of connections in ANN [11].

Several studies have been in focusing on the extension of Differential Evolution (DE) to solve multi-objective optimization problems in continuous domains. One of the first papers to explore the potential of DE for solving multi objective optimization problems (MOPs) were written by [12, 13]. In both algorithms DE is employed to create new solutions and only the non-dominated solutions are retained as the basis for the next generation. Both methods used Pareto Differential Evolution (PDE) concept as the main objective of the study.

Similar study introduced by [14] used the Pareto optimal approaches to train a multilayer perceptron network, they achieved Pareto optimal evolutionary neural network as parallel evolution of a population and considered multiple error measures as objectives. H.A. Abbass in [15] proposed multi-objective method that includes DE algorithm to train the artificial neural network and to optimize the number of hidden nodes and connection weights simultaneously. His study, benefited from the concept of multiobjective optimization and multiobjective evolutionary algorithms to evolve ANNs.

Another study by H.A. Abbass [16] that introduced multiobjective Pareto DE combined with local search algorithm for ANN to enhance the performance of algorithm. It minimizes the training error and the number of hidden nodes. This algorithm showed fast training and better generalization than traditional ANN. Similarly, in another study, the same author introduces Pareto differential evolution algorithm (PDE) hybrid with local search for evolutionary ANN to diagnose breast cancer [17] and this study has obtained promising results as well.

The work by G. P. Liu and V. Kadirkamanathan in [18] studied the benefits of multi-objective optimization for selection and identifying nonlinear systems while optimizing the size of neural networks. Instead of a single objective, three objectives were considered for performance indices or error functions.

In an attempts to optimize ANNs by multi-objective evolutionary algorithms, [8] also proposed Pareto-based multi-objective Differential Evolution algorithm that adapted to design Radial Basis Function Networks (RBFNs) hybrid with a local search for solving binary and multi class classification problems, to achieve simultaneous generalization and classification accuracy of the network. More recently, M. Cruz-Ramírez et al [19] introduced automatically designed artificial neural network and learning, the structure and weights of the ANN, for multi classification tasks in predictive microbiology using Hybrid Pareto Differential Evolution Neural Network named (HPDENN). They achieved two objectives, high classification level and highest classification level for each class.

3 Material and Methods

3.1 Differential Evolution (DE)

R. Storn and K. Price [20] introduced differential evolution (DE) algorithm for single strategy and R. Storn in [21] extended the study later to parallel computation, constraint satisfaction, constrained optimization and design centering problems. DE algorithm is an efficient population based metaheuristic search algorithm for optimization method. Recently, DE has been applied successfully in various areas and for complex non-linear problems. Due to its advantages such as simplicity, compact structure, minimal control parameters, powerful search capability and high convergence characteristics [21-24]. It has attracted a lot of researchers for its use in global optimization.

The power of DE algorithm has attracted much attention as MOEA, it has been demonstrated over the years and successfully use to solve many types of multi objective optimization problems with varying degrees of complexity and in various fields of application [25-27].

3.2 Multiobjective Differential Evolution (MODE)

A multiobjective evolutionary algorithm (MOEA), also known as multiobjective optimization algorithm (MOOA), is the process of simultaneously optimizing two or more conflicting objectives subject to certain constraints; they are a population based search. Hence, in a single run, it can get many of Pareto optimal solutions and that are attractive to this kind of algorithms. Recently, the original differential evolution algorithm DE has to be modified in order to apply the DE algorithm for multiobjective optimization problems MOPs. Presently, several previous studies have presented the prospective achievements of DE that can be an attractive alternative to extending DE for solving multi objective numerical optimization problems [12, 26, 28].

3.3 Three-Term Backpropagation Algorithm

The Three-Term Backpropagation (TBP) proposed by [29], utilizes three parameters. Beside the learning rate and a momentum factor there is third parameter which is called proportional factor (PF), this parameter was introduced in order to speed up the weight adjusting process or to increase the BP learning speed. Generally, the TBP network employs the standard architecture and procedure of the standard backpropagation algorithm that contains input layer, hidden layer and the output layer. In all layers there are number of neurons that are connected together.

4 The Proposed MODE

In recent years, the used of evolutionary computational techniques have proven themselves useful in the area of evolving artificial neural networks. For the problem of optimizing ANN, several objective functions can be considered. For instance, network accuracy, network architecture and connection weights. This section presents a MOEA evolving Artificial Neural Network (ANN). The proposed MOEA is based on the Multiobjective Differential Evolution (MODE) algorithm for Three-Term Backpropagation network (TBP). The Pareto Multi-objective Evolutionary TBP network algorithm used in this work optimizes error rates and architectures of the network simultaneously. This proposed method allows us to design TBP network, in terms of choosing the number of hidden nodes and generalizations of the network, to obtain simple and accurate TBP network. The proposed method is implemented for multi class pattern classification problems.

In this paper, MODETBP network algorithm has been proposed to determine the best performance and the corresponding architecture of the TBP network. To assist TBP network design, differential evolution (DE) and multiobjective optimization (MOO) are combined to carry out fitness evaluation and to enhance the performance capability of approximating a set of Pareto optimal solutions in a single run.

The optimization goal is to minimize the objective function. In this work, we have taken the performance of the network (Accuracy) based on the Mean Square Error (MSE) on the training set as a first objective function by minimizing the network error rate, $E=\frac{1}{N}\sum_{j=1}^{N}(t_j-o_j)^2$, where N is the number of samples. The second objective function, we have taken is to minimize the complexity of the network by optimizing the number of the nodes in the hidden layer of TBP network, $H=\sum_{h=1}^{H}\rho_h$, where, ρ_h belongs to vector ρ is the dimension of the maximum number of hidden nodes H of the network. In this objective function, the maximum number of hidden nodes H of the network as vector ρ is the binary value used to refer to the hidden node if it exists in the network or not.

4.1 Parameter Setting

The parameter setting is very important for the algorithms. This is as a result of the significant impact they have on the optimum performance. Therefore, it is required to choose the parameters carefully to find optimal values for the parameters.

In this paper, the parameters of the proposed MODE used for training the TBP network for all datasets are the same. Depending on the previous studies in literature that used, to applied DE algorithm, we chose the parameters as follow: the population size is 100, probability of crossover (C_R) used is 0.9 and the probability of mutation (F) is 0.5, while the maximum number of iterations is 1000. The fitness values are the hidden nodes and network training error or performance of the network. The training set is used to train the TBP network in order to obtain the Pareto optimal solutions, while the testing set is used to test the generalization performance of the Pareto TBP network.

5 Experimental Results

In this section we present the experimental results of the study on MODE for TBP network. The proposed method (MODETBP) is evaluated by using 10-fold cross validation technique. The results were obtained for all datasets is a Pareto optimal solution to improve the generalization on unseen data. In the experimental design, we considered three multi class datasets. Multi class pattern classification is a major classification problem with more than two different classes which accurately maps an input feature space to an output space as output, and the number of features as inputs.

Table 1 shows the dataset that was used in this study along with the following details of the dataset: number of features, number of classes and the total number of patterns. With regard to preprocessing stage, all the dataset values are normalized in the range of [0, 1].

Table 1. Summary of data sets used in the experiments

Dataset	Number of features	Number of classes	Number of patterns
Iris	4	3	150
Wine	13	3	178
Yeast	8	10	1484

For the measurements of the proposed method we used statistical measures which are Sensitivity to identify the correct positive samples, Specificity to predict the correct negative samples, accuracy to produce the level of accurate results and AUC is the area under the receiver operating characteristic curve (ROC). Since, the AUC is a portion of the area of the unit square, its value is between 0.0 and 1.0.

The experimental result presented in Table 2 shows the values of the mean and standard deviation (STD) in generalization for training and testing error rates for all runs of the experiments performed. In addition, we can easily verify that all datasets on the average as shown in Table 2, MODETBP gives promising results in both training and testing sets. Moreover, it shows that the MODETBP obtained smallest error compared with our previous method Three-Term Backpropagation network based on the Elitist Multiobjective Genetic Algorithm (MOGATBP) [30], using same dataset and objective functions.

Table 2. The training and testing error rates

		MODETBP		MOGATBP	
Dataset		***Training Error***	***Testing Error***	***Training Error***	***Testing Error***
Iris	Mean	0.1070	0.1036	0.1645	0.1654
	STD	0.0172	0.0298	0.0239	0.0224
Wine	Mean	0.1211	0.1227	0.1686	0.1682
	STD	0.0296	0.0257	0.0394	0.0433
Yeast	Mean	0.0757	0.077	0.0816	0.0816
	STD	0.0065	0.0063	0.0088	0.0088

Tables 3 present the results of the proposed method (MODETBP) and MOGATBP based on two objectives on iris, wine and yeast dataset. The Mean and SD indicate the average value and standard deviation respectively. The result of these algorithms is Pareto optimal solutions to improve the generalization on unseen data. All the results demonstrate that the MODETBP has the capability to perform better to classify the accuracy for all datasets against the MOGATBP algorithm. Additionally, Table 3 shows the statistical results for sensitivity (Sens), specificity (Spec) and Accuracy. It also gives detailed information on the MODETBP compared with MOGATBP in training and testing data. Equations 1, 2 and 3 show the calculation of those statistical measures as follows:

$$Sensitivity = \frac{TP}{TP + FN} \tag{1}$$

$$Specificity = \frac{TN}{TN + FP} \tag{2}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

Where, *TP* is true positive, *FP* is false positive, *TN* is true negative and *FN* is false negative.

The experimental result presented in Table 3 show, among other things all the datasets that were used in this study, it can be observed that MODETBP achieved better accuracy than MOGATBP. Furthermore, we can clearly notice that the result of the MODETBP obtained an accurate result of 97.02 %, 93.67 % and 90.33% for iris, wine and yeast dataset respectively.

Table 3. Accuracy, Sensitivity and Specificity

Dataset	Methods	MODETBP						MOGATBP					
	Data	*Training*			*Testing*			*Training*			*Testing*		
	Measure	*Sens%*	*Spec%*	*Accuracy %*	*Sens%*	*Spec%*	*Accuracy %*	*Sens%*	*Spec%*	*Accuracy %*	*Sens%*	*Spec%*	*Accuracy %*
Iris	Mean	96.81	97.76	97.59	93.33	96.67	97.02	34.89	99.41	78.17	34.00	99.33	77.56
	STD	0.93	1.14	2.35	6.29	3.14	2.96	27.07	1.08	8.19	24.84	2.11	7.73
Wine	Mean	94.78	97.50	95.46	93.13	95.25	93.67	23.32	98.66	73.35	99.12	74.29	74.29
	STD	5.03	0.83	2.34	7.51	3.80	5.17	35.71	2.39	10.42	2.02	11.95	11.95
Yeast	Mean	2.98	97.89	90.23	3.90	98.19	90.33	0.00	100	90.00	0.00	100	90.01
	STD	3.08	1.12	0.37	3.40	0.98	0.35	0.00	0.00	0.00	0.00	0.00	0.00

Similarly, for the sensitivity, MODETBP has achieved 93.33% for iris, 93.13% for wine and 3.90% for yeast dataset. The sensitivity of the yeast data set is very difficult, due to their unbalanced data. Furthermore, besides accuracy and sensitivity, Table 3 shows the specificity for all datasets. We can note that the specificity rate achieved is as follows: iris data has achieved 96.67%, wine data has achieved 95.25% and 98.19% obtained by yeast dataset.

In terms of mean and standard deviation, Table 3 also shows the MODETBP has produced small standard deviation for the test accuracy.

It is clearly seen that From Table 4 and Figure 1, an analysis of the accuracy and AUC compared to MEPDENf1f2 [8], MEPDENf1-f3 [8] and MOGATBP, we found that the MODETBP has the highest classification accuracy and AUC as well followed by MEPDENf1-f3 algorithm in all classification accuracy results. Both MEPDENf1f2 and MEPDENf1-f3 we compared with our method, they are a memetic multiobjective evolutionary algorithm. They use a local search method to improve the solution and achieve better accuracy of the final result. Thus it enables these methods to improve their algorithm to achieve good results. While the results of MODETBP used only multiobjective DE algorithm without the local search algorithm. In spite of this, MODETBP performs better accuracy than all algorithms.

Table 4. Comparison of the accuracy and AUC of the proposed method and other methods

Dataset	MODETBP		MOGATBP		MEPDENf1f2		MEPDENf1-f3	
	Accuracy	**AUC**	**Accuracy**	**AUC**	**Accuracy**	**AUC**	**Accuracy**	**AUC**
Iris	**97.02**	**0.95**	77.56	0.667	86.00	0.823	96.89	0.946
Wine	**93.67**	**0.942**	74.29	0.867	77.11	0.694	90.04	0.856
Yeast	**90.33**	**0.510**	90.01	0.500	90.00	0.500	90.16	0.506

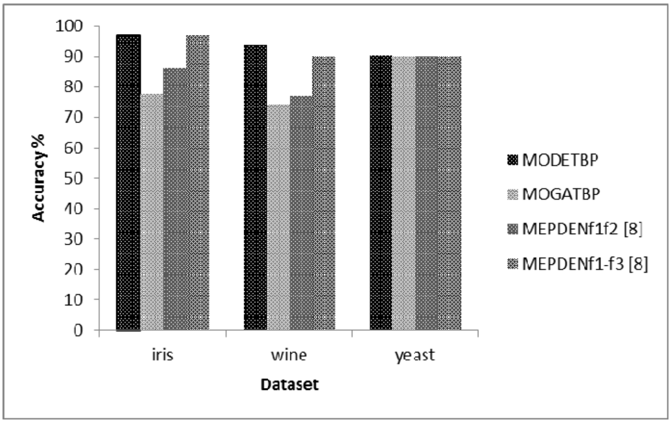


Fig. 1. Comparison of Accuracy of the MODETBP and other methods

It is clearly seen that from Table 5, the complexity of MODETBP achieved better results than all methods in all dataset when using multi objective DE algorithms. The MODETBP design smallest network architecture against all methods, except MOGATBP that used GA technique, obtained better architecture than MODETBP in both of iris and yeast, while the MODETBP performs better architecture when used with wine dataset. The number of the hidden nodes is considered one of the objectives to optimize in this study.

Table 5. Comparison of the hidden nodes of the proposed method and other methods

Dataset	MODETBP	MOGATBP	MEPDENf1f2	MEPDENf1-f3
Iris	3.8	3.6	3.9	3.9
Wine	3.8	4.6	3.8	3.8
Yeast	4.0	3.5	6.5	5.7

From Tables 4 and 5, we can conclude that MODETBP has achieved the optimization of the hidden nodes along with error rates simultaneously. It also performs well in accuracy and optimizing its generalization ability better than all methods.

6 Conclusions

This paper has presented a multiobjective evolutionary algorithm for optimizing TBP network, to achieve optimization of two objectives, which are accuracy of the network along with the complexity of the TBP network simultaneously. The proposed MODETBP algorithm has been evaluated using three types of performance evaluation indicators to assess the effect of MODETBP. In addition, MODETBP is used to develop generalization and classification accuracy for the TBP network. In this paper, an attempt was made to improve the generalization of the training and unseen data along with solving multi class pattern classification problem. The experimental results illustrate that MODETBP was able to obtain a TBP network with better classification accuracy and simpler structure compared to others.

Acknowledgements. The authors would like to thank Soft Computing Research Group (SCRG) (UTM) for the continuous support and motivation. This work is supported by a Fundamental Research Grant Scheme (FRGS - 4F347).

References

1. Mineu, N.L., Ludermir, T.B., Almeida, L.M.: Topology optimization for artificial neural networks using differential evolution. In: The 2010 International Joint Conference on Neural Networks (IJCNN), pp. 1–7 (2010)
2. Ding, S., Su, C., Yu, J.: An optimizing BP neural network algorithm based on genetic algorithm. Artificial Intelligence Review 36, 153–162 (2011)
3. Zhang, C., Shao, H., Li, Y.: Particle swarm optimisation for evolving artificial neural network. In: 2000 IEEE International Conference on Systems, Man, and Cybernetics (2000), pp. 2487–2490. IEEE (2000)
4. Ilonen, J., Kamarainen, J.-K., Lampinen, J.: Differential evolution training algorithm for feed-forward neural networks. Neural Processing Letters 17, 93–105 (2003)
5. Zhou, A., Qu, B.-Y., Li, H., Zhao, S.-Z., Suganthan, P.N., Zhang, Q.: Multiobjective evolutionary algorithms: A survey of the state of the art. Swarm and Evolutionary Computation 1, 32–49 (2011)
6. Goh, C.-K., Teoh, E.-J., Tan, K.C.: Hybrid multiobjective evolutionary design for artificial neural networks. IEEE Transactions on Neural Networks 19, 1531–1548 (2008)
7. Delgado, M., Cuéllar, M.P., Pegalajar, M.C.: Multiobjective hybrid optimization and training of recurrent neural networks. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 38, 381–403 (2008)
8. Qasem, S.N., Shamsuddin, S.M.: Memetic elitist pareto differential evolution algorithm based radial basis function networks for classification problems. Applied Soft Computing 11, 5565–5581 (2011)
9. Qasem, S.N., Shamsuddin, S.M.: Radial basis function network based on time variant multi-objective particle swarm optimization for medical diseases diagnosis. Applied Soft Computing 11, 1427–1438 (2011)
10. Fernandez Caballero, J.C., Martínez, F.J., Hervás, C., Gutiérrez, P.A.: Sensitivity versus accuracy in multiclass problems using memetic Pareto evolutionary neural networks. IEEE Transactions on Neural Networks 21, 750–770 (2010)
11. Jin, Y., Okabe, T., Sendhoff, B.: Neural network regularization and ensembling using multi-objective evolutionary algorithms. In: Congress on Evolutionary Computation, CEC 2004, pp. 1–8. IEEE (2004)
12. Abbass, H.A., Sarker, R., Newton, C.: PDE: A Pareto-frontier differential evolution approach for multi-objective optimization problems. In: Proceedings of the 2001 Congress on Evolutionary Computation, pp. 971–978. IEEE (2001)
13. Abbass, H.A., Sarker, R.: The Pareto differential evolution algorithm. International Journal on Artificial Intelligence Tools 11, 531–552 (2002)

14. Fieldsend, J.E., Singh, S.: Pareto evolutionary neural networks. IEEE Transactions on Neural Networks 16, 338–354 (2005)
15. Abbass, H.A., Sarker, R.: Simultaneous evolution of architectures and connection weights in ANNs. In: Proceedings of Artificial Neural Networks and Expert System Conference, pp. 16–21 (2001)
16. Abbass, H.A.: A memetic pareto evolutionary approach to artificial neural networks. In: Stumptner, M., Corbett, D.R., Brooks, M. (eds.) Canadian AI 2001. LNCS (LNAI), vol. 2256, pp. 1–12. Springer, Heidelberg (2001)
17. Abbass, H.A.: An evolutionary artificial neural networks approach for breast cancer diagnosis. Artificial Intelligence in Medicine 25, 265–281 (2002)
18. Liu, G., Kadirkamanathan, V.: Multiobjective criteria for neural network structure selection and identification of nonlinear systems using genetic algorithms. IEE Proceedings-Control Theory and Applications 146, 373–382 (1999)
19. Cruz-Ramírez, M., Hervás-Martínez, C., Gutiérrez, P.A., Pérez-Ortiz, M., Briceño, J., de la Mata, M.: Memetic Pareto differential evolutionary neural network used to solve an unbalanced liver transplantation problem. Soft Computing 17, 275–284 (2013)
20. Storn, R., Price, K.: Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization 11, 341–359 (1997)
21. Storn, R.: System design by constraint adaptation and differential evolution. IEEE Transactions on Evolutionary Computation 3, 22–34 (1999)
22. Brest, J., Greiner, S., Boskovic, B., Mernik, M., Zumer, V.: Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. IEEE Transactions on Evolutionary Computation 10, 646–657 (2006)
23. Rahnamayan, S., Tizhoosh, H.R., Salama, M.M.: Opposition-based differential evolution. IEEE Transactions on Evolutionary Computation 12, 64–79 (2008)
24. Tsai, J.-T., Ho, W.-H., Chou, J.-H., Guo, C.-Y.: Optimal approximation of linear systems using Taguchi-sliding-based differential evolution algorithm. Applied Soft Computing 11, 2007–2016 (2011)
25. Babu, B., Jehan, M.M.L.: Differential evolution for multi-objective optimization. In: The 2003 Congress on Evolutionary Computation, CEC 2003, pp. 2696–2703. IEEE (2003)
26. Ali, M., Siarry, P., Pant, M.: An efficient differential evolution based algorithm for solving multi-objective optimization problems. European Journal of Operational Research 217, 404–416 (2012)
27. Alatas, B., Akin, E., Karci, A.: MODENAR: Multi-objective differential evolution algorithm for mining numeric association rules. Applied Soft Computing 8, 646–656 (2008)
28. Gong, W., Cai, Z.: A multiobjective differential evolution algorithm for constrained optimization. In: IEEE Congress on Evolutionary Computation, CEC 2008 (IEEE World Congress on Computational Intelligence), pp. 181–188. IEEE (2008)
29. Zweiri, Y., Whidborne, J., Seneviratne, L.: A three-term backpropagation algorithm. Neurocomputing 50, 305–318 (2003)
30. Ibrahim, A.O., Shamsuddin, S.M., Ahmad, N.B., Qasem, S.N.: Three-Term Backpropagation Network Based On Elitist Multiobjective Genetic Algorithm for Medical Diseases Diagnosis Classification. Life Science Journal 10 (2013)

Ontology Development to Handle Semantic Relationship between Moodle E-learning and Question Bank System

Arda Yunianta[1,2] , Norazah Yusof[1,*], Herlina Jayadianti[3],
Mohd Shahizan Othman[1], and Shaffika Suhaimi[1]

[1] Faculty of Computer Science and Information System,
Universiti Teknologi Malaysia, 81310, Johor Malaysia
{yarda2,sshaffika2}@live.utm.my,
{shahizan,norazah}@utm.my
[2] Faculty of Information Technology and Communication,
Mulawarman University, 75119,
Samarinda Kalimantan Timur, Indonesia
arda.aldoe00@gmail.com
[3] Faculty of Industrial Technology,
Universitas Pembangunan Nasioanal,
55283, Yogyakarta, Indonesia
herlinajayadianti@gmail.com

Abstract. Distributed and various systems on learning environment produce heterogeneity data in data level implementation. Heterogeneity data on learning environment is about different data representation between learning system. This problem makes the integration problem increasingly complex. Semantic relationship is a very interesting issue in learning environment case study. Difference data representation on each data source makes numerous systems difficult to communicated and integrated with the others. Many researchers found that the semantic technology is the best way to resolve the heterogeneity data representation issues. Semantic technology can handle heterogeneity of data, data with different representations in different data sources. Semantic technology also can do data mapping from different database and different data format that have same meaning data. This paper focuses on semantic data mapping to handle the semantic relationship on heterogeneity data representation using semantic ontology approach. In the first level process, using D2RQ engine to produce turtle (.ttl) file format that can be used for Local Java Application using Jena Library and Triple Store. In the second level process we develop ontology knowledge using protégé tools to handle semantic relationship. In this paper, produce ontology knowledge to handle a semantic relationship between Moodle E-learning system and Question Bank system.

Keywords: D2RQ, Data Integration, Heterogeneity Data, Learning Environment, Ontology, Semantic Mapping.

[*] Corresponding author.

T. Herawan et al. (eds.), *Recent Advances on Soft Computing and Data Mining SCDM 2014*, Advances in Intelligent Systems and Computing 287,
DOI: 10.1007/978-3-319-07692-8_65, © Springer International Publishing Switzerland 2014

1 Introduction

The heterogeneity of data is a common phenomenon in distributed information sources and it is growing with the development of computer and information technologies that have created a huge amount of data and information [1],[2]. Heterogeneity of data, data with different representations and sources, are the other problem existing in current obsolescence management tools, also data conflicts are more common than data agreement [3],[4]. At the same time today's software systems develop into more distributed and more autonomous. Both these trends are a natural reason for the intense efforts in a domain of data integration.

Heterogeneity on learning environment is about different data and applications to support a learning process in some education institutions. Different applications are develop for specific purposes based on function and feature that included on that applications [5]. Start from application to support learning activity between lecturers and students calls e-learning, student financial application, until student grading application. Different application system with numerous and heterogeneity information, data sources, databases system and data representation makes communication and integration process between this applications difficult to implemented.

Nowadays learning environment is becoming popular because of their convenience and accessibility to help and support learning process [6],[7]. Data integration process between applications on learning environment to be an important part to gain learning knowledge that can support decision making process on executive level on the organization. Implementation of data integration still has a many problems to be solved. Exchanging and merging data from loosely coupled, heterogeneous data representation and mapping data on different data source are the serious problem on data integration process [8]-[14].

A lot of application integrations are implemented in the current days. Enterprise Application Integration (EAI) is the one of famous integration application that implemented in current days. EAI enables the enterprise to function more efficiently, provide better services for its customers and to ensure faster realization of its business ideas. It also ensures quicker and more reliable communication of business information that supports the strategic and tactical business goals [15]. Enterprise Information Integration (EII) is the other data integration application that already implemented in many organizations. EII is based on service oriented architecture to implement the integration process [16].

Researchers are using Semantic ontologies extensively in semantic data mapping approach to annotate their data, to drive decision-support systems, to integrate data, and to perform natural language processing and information extraction. Ontologies provide a means of formally specifying complex descriptions and information about relationships in a way that is expressive yet amenable to automated processing and reasoning [17]-[19]. As such, they offer the promise of facilitated information sharing, data fusion and exchange among many, distributed and possibly heterogeneous data sources [4].

However, the focus of this paper is to produce data source mapping files between Moodle e-learning system and question bank system to handle the heterogeneity problem and to create semantic relationship on ontology knowledge. In the future, this semantic mapping will be integrated with the other learning system to communicate and collaboration on specific data that have the same meaning and semantic relationship to produce Decision Support System for executive level in organization.

In this paper, we produce ontology knowledge between moodle e-learning and question bank system with several parts process. In the first process, semantic data mapping process using D2RQ engine will produce data mapping language with turtle (.ttl) file format that can be used for Local Java Application using Jena Library and Triple Store. In the second process, is to develop ontology knowledge using protégé software to produce ontology knowledge that can be used together with turtle file to produce semantic data integration approach.

2 Semantic Data Integration Method

2.1 Data Mapping Schema

In generally, semantic data mapping is the relationship between four parts that are important parts on semantic data mapping and integration data. The core part is semantic data mapping that will handle communication and integration with the other three parts. Second part is e-learning and question bank data source that will be mapping in semantic data mapping. The third part is a local application that using semantic data mapping. And the fourth part is the other system that will be communicated and integrated from outside environment using HTTP Protocol. Semantic data mapping architecture can be seen in fig. 1.

The mapping defines a virtual RDF graph that contains information from the database. This is similar to the concept of views in SQL, except that the virtual data structure is an RDF graph instead of a virtual relational table. The virtual RDF graph can be accessed in various ways, depending on what's offered by the implementation. The D2RQ Platform provides SPARQL access, a Linked Data server, an RDF dump generator, a simple HTML interface, and Jena API access to D2RQ-mapped databases [17].

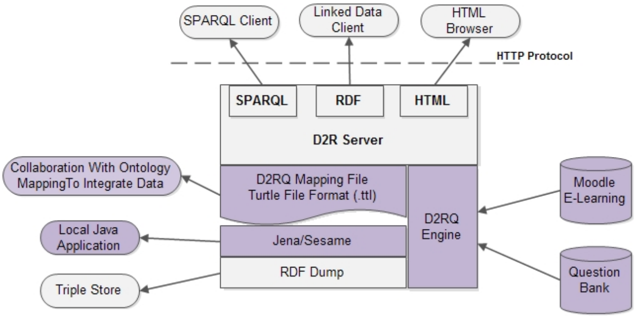


Fig. 1. Semantic Data Mapping Schema

In the semantic data mapping, there are three important parts that we can see in figure 1. The first part is D2RQ engine that is the core part in semantic data mapping process. D2RQ engine is responsible to communicate with a local data source and produce D2RQ data mapping file that can be used to communicate with local application using jena library and RDF Dump. The second part is a D2R server to communicate and integrate with the others system from outside environment using HTTP Protocol. In this part will produce SPARQL that can be access from SPRQL Clients, RDF that can be accessed from linked data clients and HTML that can be accessed from HTML browser [20].

In the third part is D2RQ data mapping file is a text mode file with turtle file format (.ttl) that contain data mapping from a local data source based on ontology based language. The D2RQ Mapping Language is a declarative language for describing the relation between a relational database schema and RDFS vocabularies or OWL ontologies. A D2RQ mapping is itself an RDF document written in Turtle syntax. The mapping is expressed using terms in the D2RQ namespace. A namespace is a domain that serves to guarantee the uniqueness of identifiers. Written like uniform resource locator (URL), example http://www.wiwiss.fu-berlin.de/suhl/bizer/D2RQ/0.1#. The terms in this namespace are formally defined in the D2RQ RDF schema (Turtle version, RDF/XML version).

Implementation for this research is focus on utilization D2RQ Engine to produce turtle file format to collaborate with ontology mapping using local java application with Jena as a library support to make semantic data mapping between moodle e-learning and question bank system.

2.2 Heterogeneity Data Sources Representation

In this paper, we integrate two data source between moodle e-learning and question bank system. Between two systems are interconnected information, which have a semantic relationship.

E-learning system is a tool system contains learning management systems to support learning activities such as courses, assignment, quiz, forum and the others online interactive classes between lecturer and students. E-learning is increasingly being used in commercial organizations to improve efficiency and reduce costs, and also being adopted and integrated with the others system in their environment [21],[22]. A lot of tables on database source on moodle e-learning system but only a few tables will be used to perform semantic data mapping with question bank data source that have a semantic relationship to implement using semantic technology approach.

The related tables are used to implement this research is a tables containing the lecturer activities conducted in the E-learning system. The lecturer activities are assignment, quizzes, lab activities, project and presentation saved in five tables in the Moodle data source. The five tables are *mdl_assign, mdl_quiz, mdl_workshop, mdl_page dan mdl_label*. Figure 2 shows five tables are used in Moodle data source.

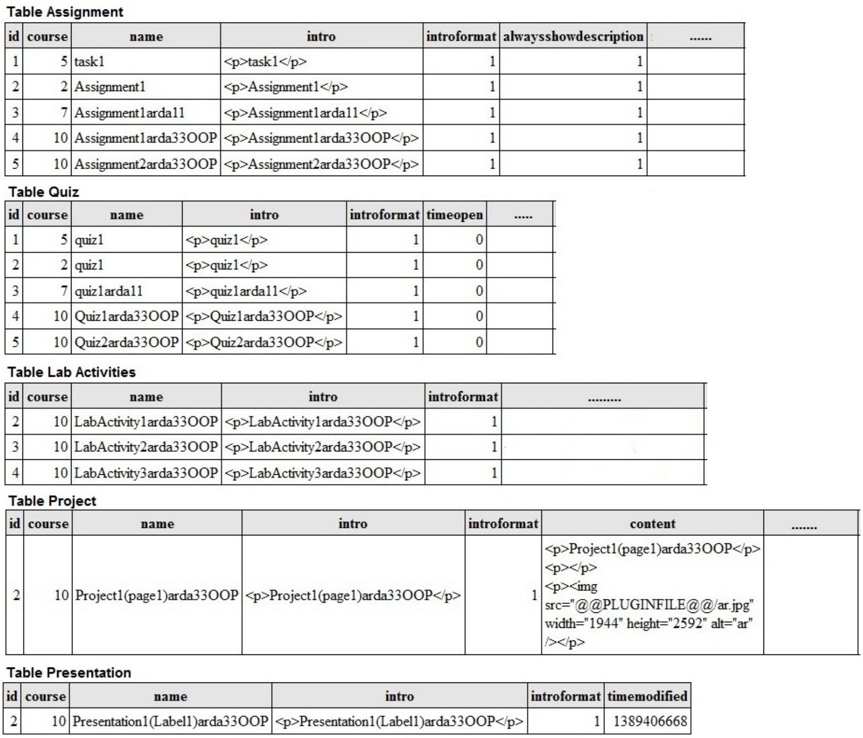


Table Assignment

id	course	name	intro	introformat	alwaysshowdescription	......
1	5	task1	<p>task1</p>	1	1	
2	2	Assignment1	<p>Assignment1</p>	1	1	
3	7	Assignment1arda11	<p>Assignment1arda11</p>	1	1	
4	10	Assignment1arda33OOP	<p>Assignment1arda33OOP</p>	1	1	
5	10	Assignment2arda33OOP	<p>Assignment2arda33OOP</p>	1	1	

Table Quiz

id	course	name	intro	introformat	timeopen	.....
1	5	quiz1	<p>quiz1</p>	1	0	
2	2	quiz1	<p>quiz1</p>	1	0	
3	7	quiz1arda11	<p>quiz1arda11</p>	1	0	
4	10	Quiz1arda33OOP	<p>Quiz1arda33OOP</p>	1	0	
5	10	Quiz2arda33OOP	<p>Quiz2arda33OOP</p>	1	0	

Table Lab Activities

id	course	name	intro	introformat	....
2	10	LabActivity1arda33OOP	<p>LabActivity1arda33OOP</p>	1	
3	10	LabActivity2arda33OOP	<p>LabActivity2arda33OOP</p>	1	
4	10	LabActivity3arda33OOP	<p>LabActivity3arda33OOP</p>	1	

Table Project

id	course	name	intro	introformat	content	....
2	10	Project1(page1)arda33OOP	<p>Project1(page1)arda33OOP</p>	1	<p>Project1(page1)arda33OOP</p> <p></p> <p><img src="@@PLUGINFILE@@/ar.jpg" width="1944" height="2592" alt="ar" /></p>	

Table Presentation

id	course	name	intro	introformat	timemodified
2	10	Presentation1(Label1)arda33OOP	<p>Presentation1(Label1)arda33OOP</p>	1	1389406668

Fig. 2. Shows Some Parts of The Moodle Database Tables

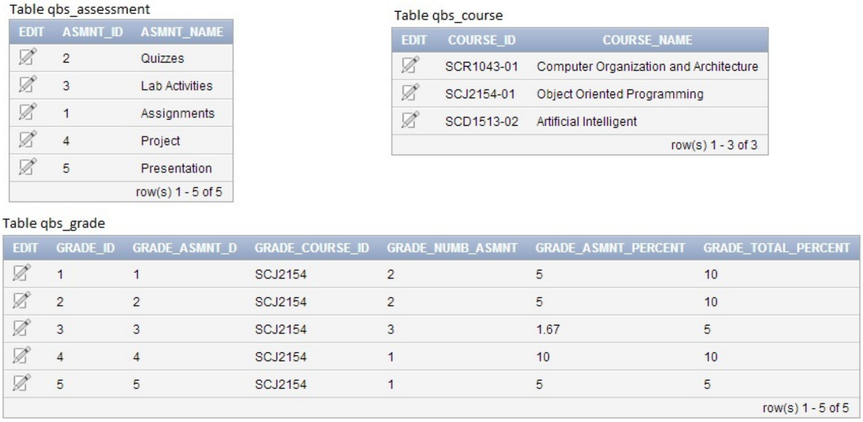


Table qbs_assessment

EDIT	ASMNT_ID	ASMNT_NAME
✎	2	Quizzes
✎	3	Lab Activities
✎	1	Assignments
✎	4	Project
✎	5	Presentation

row(s) 1 - 5 of 5

Table qbs_course

EDIT	COURSE_ID	COURSE_NAME
✎	SCR1043-01	Computer Organization and Architecture
✎	SCJ2154-01	Object Oriented Programming
✎	SCD1513-02	Artificial Intelligent

row(s) 1 - 3 of 3

Table qbs_grade

EDIT	GRADE_ID	GRADE_ASMNT_D	GRADE_COURSE_ID	GRADE_NUMB_ASMNT	GRADE_ASMNT_PERCENT	GRADE_TOTAL_PERCENT
✎	1	1	SCJ2154	2	5	10
✎	2	2	SCJ2154	2	5	10
✎	3	3	SCJ2154	3	1.67	5
✎	4	4	SCJ2154	1	10	10
✎	5	5	SCJ2154	1	5	5

row(s) 1 - 5 of 5

Fig. 3. Question Bank Tables

Question bank is a system that can be manages lecturer activities to conduct learning process. This project also emphasizes on the outcome based learning approach, in which the question items are categorized based on the cognitive level of Bloom's taxonomy, as well as the learning objectives. Question bank system also allows lecturers to prepare questions for various evaluation purposes such as quizzes, tests and final examination. The system will generate a set of exam paper and export to the doc format. In this system, there is information about assessment activities as a standard set by institution to make a learning process.

The related tables are used in the Question bank system is a table containing the assessment schema on each subject course. Tables are used to implement this research are table *qbs_assessment, table qbs_course and table qbs_grade*. Detail tables can be seen in Figure 3.

3 Heterogeneity Data Mapping Using Ontology

The overall mapping process is starts from D2RQ engine that have function to map from database table schema to XML file format that call turtle file. This process produces two files that can be combined to one turtle file to use in main process on a semantic data mapping step.

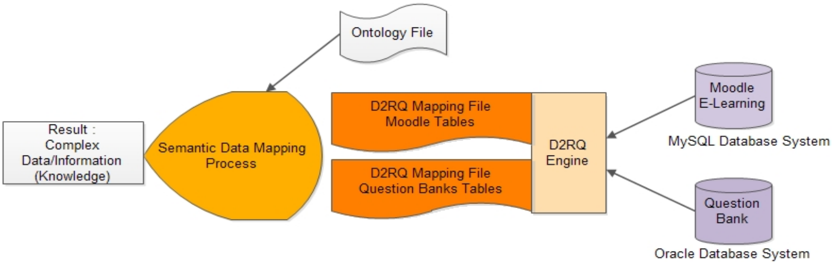

Ontology File
Result : Complex Data/Information (Knowledge)
Semantic Data Mapping Process
D2RQ Mapping File Moodle Tables
D2RQ Mapping File Question Banks Tables
D2RQ Engine
Moodle E-Learning
MySQL Database System
Question Bank
Oracle Database System

Fig. 4. Database Mapping Process

Result from this process is lecturer activities recorded in the moodle system and data about assessment activities that must be done in the learning process. From this implementation we want to monitor from lecturer sides, whether they perform in accordance with the curriculum that have been set.

3.1 Database Mapping Process

In this step, we want to produce data mapping file from two databases, moodle e-learning and question bank system. D2RQ is a tool to semi automation mapping process from database table schema to XML format calls Turtle file.

```
@prefix map: <#> .
@prefix db: <> .
@prefix vocab: <vocab/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix d2rq: <http://www.wiwiss.fu-berlin.de/suhl/bizer/D2RQ/0.1#> .
@prefix jdbc: <http://d2rq.org/terms/jdbc/> .
@prefix moodle: <http://www.utm.my/mapping/moodle#> .

map:database a d2rq:Database;
        d2rq:jdbcDriver "com.mysql.jdbc.Driver";
        d2rq:jdbcDSN "jdbc:mysql://localhost/moodle23";
        d2rq:username "root";
        jdbc:autoReconnect "true";
        jdbc:zeroDateTimeBehavior "convertToNull";
        .
# Table mdl_assign
map:mdl_assign a d2rq:ClassMap;
        d2rq:dataStorage map:database;
        d2rq:uriPattern
"http://www.utm.my/mapping/moodle#mdl_assign/@@mdl_assign.id@@";
        d2rq:class vocab:mdl_assign;
        d2rq:classDefinitionLabel "mdl_assign";
        .
map:mdl_assign__label a d2rq:PropertyBridge;
        d2rq:belongsToClassMap map:mdl_assign;
        d2rq:property rdfs:label;
        d2rq:pattern "mdl_assign #@@mdl_assign.id@@";
        .
map:mdl_assign_id a d2rq:PropertyBridge;
        d2rq:belongsToClassMap map:mdl_assign;
        d2rq:property vocab:mdl_assign_id;
        d2rq:propertyDefinitionLabel "mdl_assign id";
        d2rq:column "mdl_assign.id";
        d2rq:datatype xsd:integer;
……
```

Fig. 5. Moodle Mapping File

These files will describe all resources to explain a mapping process. Start with URI description as a domain that serves to guarantee the uniqueness of identifiers. URI description of these files is “moodle: <http://www.utm.my/mapping/moodle#>”. The next line is to describe a database connection to get database and table mapping from database system. After describe database connection, the next lines is the main part of this files is to map from the tables schema into the ontology knowledge. These files can be merged into one file turtle that contain two database mapping description to use in the semantic mapping process.

```
@prefix map: <#> .
@prefix db: <> .
@prefix vocab: <vocab/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix d2rq: <http://www.wiwiss.fu-berlin.de/suhl/bizer/D2RQ/0.1#> .
@prefix jdbc: <http://d2rq.org/terms/jdbc/> .
@prefix moodle: <http://www.utm.my/mapping/moodle#> .

map:database a d2rq:Database;
        d2rq:jdbcDriver "oracle.jdbc.OracleDriver";
        d2rq:jdbcDSN "jdbc:oracle:thin:@//localhost:1521/xe";
        d2rq:username "QBS";
        d2rq:password "mypassword";
        .
# Table QBS.QBS_ASSESSMENT
map:QBS_QBS_ASSESSMENT a d2rq:ClassMap;
        d2rq:dataStorage map:database;
        d2rq:uriPattern
""http://www.utm.my/mapping/moodle#QBS/QBS_ASSESSMENT/@@QBS.QBS_ASSESS
MENT.ASMNT_ID@@";
        d2rq:class vocab:QBS_QBS_ASSESSMENT;
        d2rq:classDefinitionLabel "QBS.QBS_ASSESSMENT";
        .
map:QBS_QBS_ASSESSMENT__label a d2rq:PropertyBridge;
        d2rq:belongsToClassMap map:QBS_QBS_ASSESSMENT;
        d2rq:property rdfs:label;
        d2rq:pattern "QBS_ASSESSMENT #@@QBS.QBS_ASSESSMENT.ASMNT_ID@@";
        .
map:QBS_QBS_ASSESSMENT_ASMNT_ID a d2rq:PropertyBridge;
        d2rq:belongsToClassMap map:QBS_QBS_ASSESSMENT;
        d2rq:property vocab:QBS_QBS_ASSESSMENT_ASMNT_ID;
        d2rq:propertyDefinitionLabel "QBS_ASSESSMENT ASMNT_ID";
        d2rq:column "QBS.QBS_ASSESSMENT.ASMNT_ID";
        d2rq:datatype xsd:decimal;
        .
map:QBS_QBS_ASSESSMENT_ASMNT_NAME a d2rq:PropertyBridge;
        d2rq:belongsToClassMap map:QBS_QBS_ASSESSMENT;
        d2rq:property vocab:QBS_QBS_ASSESSMENT_ASMNT_NAME;
        d2rq:propertyDefinitionLabel "QBS_ASSESSMENT ASMNT_NAME";
        d2rq:column "QBS.QBS_ASSESSMENT.ASMNT_NAME";
…….
```

Fig. 6. QBS Mapping File

3.2 Ontology Data Mapping Visualization

This is an ontology class diagram to visualize the ontology file using Protégé tool. This ontology consists of three main classes are *Lecturer, SubjectCourse, AssessmentSchema* and *LecturerActivities*. Start from *Lecturer* and *SubjectCourse* class have an instance as an individual from each class. Lecturer class have two instances are *ArdaYunianta* and *NorazahYusof*. And *SubjectCourse* class has three instances are *ArtificialIntelligent, ComputerOrganizayionAndArchitecture* and *ObjectOrientedProgramming*. The detailed ontology class diagram can be seen in figure 7.

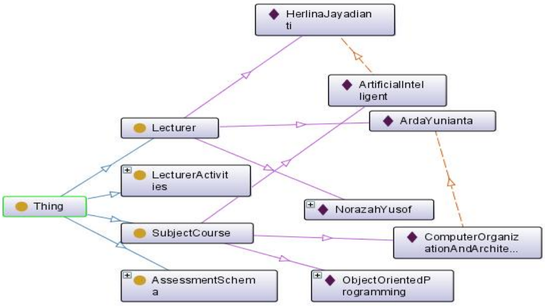


Fig. 7. Lecturer and *SubjectCourse* instance class

The other classes are *LecturerActivities* and *AssessmentSchema*. *LecturerActivities* contains seven subclasses are *AssignmentAct, QuizAct, LabActivityAct, ProjectAct, PresentationAct, MidExamAct* and *FinalExamAct* class. Whereas for *AssessmentSchema* have two subclasses been Number and Percentage class. The detailed ontology class diagram can be seen in figure 8.

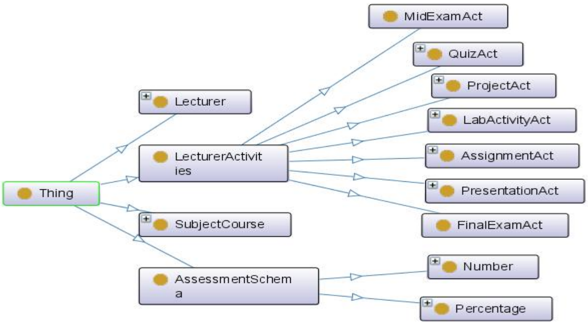


Fig. 8. LecturerActivities and AssessmentSchema Subclass

After describing all classes and instances contain on ontology knowledge, now it is time to describe semantic relationship that occurred in the ontological knowledge. There are sixteen semantic relationships on this ontology knowledge are *hasLecturer, perform, hasNumberOfAssignment, hasNumberOfQuizzes, hasNumberOfLabActivity, hasNumberOfProject, hasNumberOfPresentation, hasNumberOfMidExam, hasNumberOfFinalExam, hasPercentageOfAssignment, hasPercentageOfQuiz, hasPercentageOfLabActivity, hasPercentageOfProject, hasPercentageOfPresentation, hasPercentageOfMidExam* and *hasPercentageOfFinalExam*. The detailed semantic relationship on ontology knowledge can be seen on figure 9.

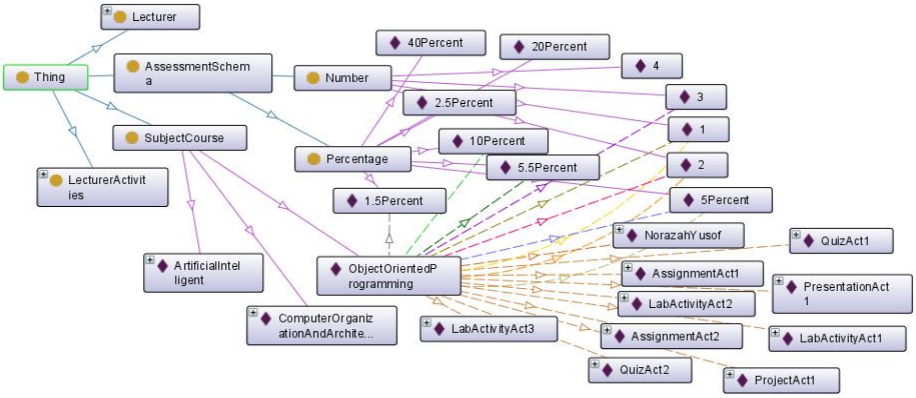


Fig. 9. Semantic Relationship on Ontology

4 Conclusion and Future Work

Semantic approach is the best way to handle the heterogeneity data representation that has a semantic relationship between data sources. Semantic technology builds a new knowledge that cannot be resolved on existing data integration system. Semantic data source mapping and ontology development will be a part of semantic data integration process to produce new information from several data sources. Implementation from this research produces a solution to solve heterogeneity issues on data representation level and semantic relationship issues between numerous data sources on learning environment. In this paper we have developed ontology knowledge that contains sixteen semantic relationships are *hasLecturer, perform, hasNumberOfAssignment, hasNumberOfQuizzes, hasNumberOfLabActivity, hasNumberOfProject, hasNumberOfPresentation, hasNumberOfMidExam, hasNumberOfFinalExam, hasPercentageOfAssignment, hasPercentageOfQuiz, hasPercentageOfLabActivity, hasPercentageOfProject, hasPercentageOfPresentation, hasPercentageOfMidExam* and *has-PercentageOfFinalExam.*

References

1. Kashyap, V., Sheth, A.: Semantic heterogeneity in global information systems: The role of metedata, context and ontologies. In: Papazoglou, M.P., Schlageter, G. (eds.) Cooperative Information Systems, pp. 139–178. Academic Press, San Diego (1997)
2. Kim, W., Seo, J.: Classifying schematic and data heterogeneity in multi database systems. IEEE Computer 24(12), 12–18 (1991)
3. Sandborn, P., Terpenny, J., Rai, R., Nelson, R., Zheng, L., Schafer, C.: Knowledge representation and design for managing product obsolescence. In: Proceedings of NSF Civil, Mechanical and Manufacturing Innovation Grantees Conference, Atlanta, Georgia (2011)

4. LePendu, P., Dou, D.: Using ontology databases for scalable query answering, inconsistency detection, and data integration 37, 217–244 (2011)
5. Shyamala, R., Sunitha, R., Aghila, G.: Towards Learner Model Sharing Among Heterogeneous E-Learning Environments. International Journal of Engineering Science and Technology (IJEST) 3(4), 2034–(2040)
6. Liu, X., Saddik, A.E., Georganas, N.D.: An Implementable Architecture of an E-Learning System. In: CCECE 2003–CCGEI 2003, Montreal (2003)
7. Dietinger, T.: Aspects of E-learning Environments. PhD thesis, Graz University of Technology (2003)
8. Arenas, M., Libkin, L.: XML Data Exchange: Consistency and Query Answering. In: Proc. of the 24th ACM SIGMOD Symposium on Principles of Database Systems, PODS 2005. ACM (2005)
9. Bonifati, A., Chrysanthis, P., Ouksel, A., Satter, K.-U.: Distributed Databases and Peer-to-Peer Databases: Past and Present. SIGMOD Record 37, 1 (2008)
10. Bouquet, P., Serafini, L., Zanobini, S.: Peer-to-peer semantic coordination. Journal of Web Semantics 2(1), 81–97 (2004)
11. Calvanese, D., Giacomo, G., Lenzerini, M., Rosati, R.: Logical Founda-tions of Peer-To-Peer Data Integration. In: Proc. of the 23rd ACM SIGMOD Symposium on Principles of Database Systems, PODS 2004, pp. 241–251. ACM (2004)
12. Fagin, R., Kolaitis, P., Popa, L.: Data exchange: getting to the core. ACM Trans. Database Syst. 30(1) (2005)
13. Pankowski, T.: Management of executable schema mappings for XML data exchange. In: Grust, T., et al. (eds.) EDBT 2006. LNCS, vol. 4254, pp. 264–277. Springer, Heidelberg (2006)
14. Pankowski, T.: XML data integration in SixP2P - a theoretical framework. In: Data Management in P2P Systems, pp. 11–18. ACM (2008)
15. Ana, C., Kresimir, F.: EAI issues and best practices. In: Proceedings of the 9th WSEAS International Conference on Applied Computer Science, pp. 135–139 (2009)
16. Kong, Z., Wang, D., Zhang, J.: A Strategic Framework for Enterprise Information Integration of ERP and E-Commerce. In: Xu, L.D., Min Tjoa, A., Chaudhry, S.S. (eds.) Research and Practical Issues of Enterprise Information Systems II. IFIP, vol. 254, pp. 701–705. Springer, Boston (2007)
17. Bellatreche, L., Dung, N.X., Pierra, G., Hondjack, D.: Contribution of ontology-based data modeling to automatic integration of electronic catalogues within engineering databases. Computers in Industry 57 (2006)
18. Castano, S., Antonellis, V., Vimercati, S.D.C.: Global viewing of heterogeneous data sources. IEEE Transactions on Knowledge and Data Engineering 13(2), 277–297 (2001)
19. Chen, Y.: Knowledge integration and sharing for collaborative molding product design and process development. Computers in Industry 61, 659–675 (2010)
20. Cyganiak, R., Bizer, C., Garbers, J., Maresch, O., Becker, C.: The D2RQ Mapping Language. v0.8 – 2012-03-12 (2012) (February 2014) (retrieved)
21. Carnevale, D.: It's Education Online. It's Someplace You Aren't. What's It Called? The Chronicle of Higher Education 47(18), 33–37 (2001)
22. Ning, Z., Bao, H.: Research on E-learning with Digital Technology in Distance Education. Paper presented at the International Conference on e-Education, e-Business, e-Management, and e-Learning, IC4E 2010, January 22-24, pp. 299–302 (2010)

Author Index