Advances in Intelligent Systems and Computing 723

Aboul Ella Hassanien
Mohamed F. Tolba
Mohamed Elhoseny
Mohamed Mostafa *Editors*

The International Conference on Advanced Machine Learning Technologies and Applications (AMLTA2018)

Springer

Advances in Intelligent Systems and Computing

Volume 723

Series editor

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

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.

More information about this series at http://www.springer.com/series/11156

Aboul Ella Hassanien · Mohamed F. Tolba
Mohamed Elhoseny · Mohamed Mostafa
Editors

The International Conference on Advanced Machine Learning Technologies and Applications (AMLTA2018)

Springer

Editors
Aboul Ella Hassanien
Faculty of Computers and Information,
Information Technology Department
Cairo University
Giza
Egypt

Mohamed F. Tolba
Faculty of Computers and Information
Sciences
Ain Shams University
Cairo
Egypt

Mohamed Elhoseny
Faculty of Computers and Information,
Department of Computer Science
and Engineering
Mansoura University
Dakahlia
Egypt

Mohamed Mostafa
Arab Academy for Science, Technology
and Maritime Transport (AASTMT)
Dokki
Egypt

ISSN 2194-5357 ISSN 2194-5365 (electronic)
Advances in Intelligent Systems and Computing
ISBN 978-3-319-74689-0 ISBN 978-3-319-74690-6 (eBook)
https://doi.org/10.1007/978-3-319-74690-6

Library of Congress Control Number: 2018930747

Printed on acid-free paper

This Springer imprint is published by the registered company Springer International Publishing AG
part of Springer Nature
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

This volume constitutes the refereed proceedings of the Third International Conference on Advanced Machine Learning Technologies and Applications, AMLTA 2018, held in Cairo, Egypt, in February 22–24, 2018.

In response to the call for papers for AMLTA2018, 159 papers were submitted for presentation and inclusion in the proceedings of the conference.

After a careful blind refereeing process, 68 papers were selected for inclusion in the conference proceedings. The papers were evaluated and ranked on the basis of their significance, novelty, and technical quality by at least two reviewers per paper. After a careful blind refereeing process, 68 papers were selected for inclusion in the conference proceedings. The papers cover current research in machine learning, big data, Internet of Things, biomedical engineering, fuzzy log, security, and intelligence swarms and optimization. Also, one workshop organized by Prof. Roheet Bhatnagar (Manipal University Jaipur, Rajasthan, India) tilled Soft Computing Applications in Data Science (SCADS). In addition to these papers, the program included one keynote talk by Prof. Qing Tan, (University Drive, Athabasca, Alberta, Canada) the talk title is Big Data Privacy- Is Blockchain an Exit?

We express our sincere thanks to the plenary speakers, workshop chairs, and international program committee members for helping us to formulate a rich technical program. We would like to extend our sincere appreciation for the outstanding work contributed over many months by the Organizing Committee: Local Organization Chair, and Publicity Chair. We also wish to express our appreciation to the SRGE members for their assistance. We would like to emphasize that the success of AMLTA2018 would not have been possible without the support of many committed volunteers who generously contributed their time, expertise, and resources toward making the conference an unqualified success. Finally, thanks to Springer team for their supporting in all stages of the production of the proceedings. We hope that you will enjoy the conference program.

Aboul Ella Hassanien
Mohamed F. Tolba
Mohamed Elhoseny
Mohamed Mostafa Fouad

Organization

Honorary Chair

Mohamed Jemni University of Tunis, Director of ICT at ALECSO

General Chair

M. F. Tolba, Egypt

International Advisory Board

Fahmy Tolba, Egypt
Mahmoud Abdel-Aty Hasoob, Egypt
Amin Shoukry, Egypt
Roheet Bhatnagar, India
A. Sharaf Eldin, Egypt
Ajith Abraham, USA
Siddhartha Bhattacharyya, India
Vaclav Snasel, Czech Republic
Dominik Slezak, Poland
Janusz Kacprzyk, Poland
Tai-hoon Kim, Korea
Qing Tan, Canada
Hiroshi Sakai, Japan

Program Chairs

Aboul Ella Hassanien, Egypt
Mostafa Mostafa Hashim, Singapore

Publicity Chairs

Mohamd Mostafa Fouad, Egypt
Saurav Karmakar, India
Ashraf Darwish, Egypt
Nour Mahmoud, Egypt
Mohamed Elhoseny, USA
J. Amudhavel, India

Technical Program Committee

Hani M. K. Mahdi, Egypt
Essam Elfakharany, Egypt
Ashraf AbdelRaouf, Egypt
Aarti Singh, India
Tahani Alsubait, UK
Evgenia Theodotou, Greece
Pavel Kromer, Czech Republic
Irma Aslanishvili, Czech Republic
Jan Platos, Czech Republic
Ivan Zelinka, Czech Republic
Sebastián Basterrech, Czech Republic
Natalia Spyropoulou, Greece
Dimitris Sedaris, Greece
Vassiliki Pliogou, Greece
Pilios Stavrou, Greece
Eleni Seralidou, Greece
Stelios Kavalaris, Greece
Litsa Charitaki, Greece
Elena Amaricai, Greece
Qing Tan, Canada
Pascal Roubides, USA
Manal Abdullah, KSA
Mandia Athanasopoulou, Greece
Vicky Goltsi, Greece

Mohammad Reza Noruzi, Iran
Abdelhameed Ibrahim, Egypt
Ahmad Taher Azar, Egypt
Ahmed Anter, Egypt
Ahmed Elhayek, Germany
Alaa Tharwat, Germany
Amira S. Ashour, KSA
Edgard Marx, Germany
Islam Amin, Australia
Ivan Ermilov, Germany
Mahmoud Awadallah, USA
Minu Kesheri, India
Mohammed Abdel-Megeed, Egypt
Mona Solyman, Egypt
Nabiha Azizi, Algeria
Namshik Han, UK
Noreen Kausar, KSA
Noura Semary, Egypt
Rania Hodhod, USA
Reham Ahmed, Egypt
Sara Abdelkader, Canada
Sayan Chakraborty, India
Shoji Tominaga, Japan
Siva Ganesh Malla, India
Soumya Banerjee, India
Sourav Samanta, India
Suvojit Acharjee, India
Swarna Kanchan, India
Takahiko Horiuchi, Japan
Tommaso Soru, Germany
Wahiba Ben Abdessalem, KSA
Zeineb Chelly, Tunis

Local Arrangement Chairs

Essam Halim Houssein (Chair), Egypt
Ahmed Talaat, Egypt
Mohamed Abd Elfattah, Egypt

Contents

Swarm Intelligence and Optimization

Fuzzy Logic and Applications

Advanced Machine Learning and Applications

Big Data and Classification

Intelligent Systems and Applications

Swarm Intelligence and Optimization

A Hybrid Grey Wolf-Bat Algorithm for Global Optimization

Mohammed ElGayyar[1](✉), E. Emary[2], N. H. Sweilam[1], and M. Abdelazeem[3]

[1] Department of Mathematics, Faculty of Science, Cairo University, Giza, Egypt
{melgayar,nsweilam}@sci.cu.edu.eg

[2] Faculty of Computers and Information, Cairo University, Giza, Egypt
eidemary@yahoo.com

[3] National Research Institute of Astronomy and Geophysics (NRIAG), Helwan, Egypt
maazeem03@yahoo.com

Abstract. Bio-inspired algorithms are now becoming powerful methods for solving many real-world optimization problems. In this paper, we propose a hybrid approach involving Grey Wolf optimizer (GWO) and Bat swarm optimizer (BA) for global function optimization problems. GWO is well known for its balanced exploration/exploitation behavior, while BA is known to be more exploitative due to its low exploration ability in some conditions. We use GWO exploration skills to explore the search space effectively and BA local search capabilities to refine the solution. In our hybrid algorithm, namely (GWOBA), GWO is used to explore the problem space alone and pass the best two solutions to BA to guide its local search, then BA digs deeper and find the best solution. The new proposed approach has been tested using 30 standard benchmark functions from CEC2017 benchmark suite. The performance of the hybrid algorithm has been compared to the original GWO, BA and the Whale optimization algorithm (WOA). We use a set of performance indicators to evaluate the efficiency of the method. Results over various dimensions show the superiority of the proposed algorithm.

Keywords: Optimization · Greywolf optimizer · Bat algorithm CEC2017 · Hybridization

1 Introduction

In most real-world applications, there is always a need to minimize (cost, time or waste) or maximize (performance, benefits or profit). Both minimization and maximization are usually considered the main objectives of optimization problems. In many cases, traditional deterministic algorithms fail to solve optimization problems in practice. Real problems are usually nonlinear, complex and multimodal with many local optima, they might also suffer from complex constraints

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 3–12, 2018.
https://doi.org/10.1007/978-3-319-74690-6_1

and high number of dimensions, which motivated researchers and scientists to design non-deterministic optimization algorithms to solve such problems.

Almost all new optimization algorithms are inspired from Nature, hence they are referred to as nature-inspired algorithms. Some algorithms were inspired from chemical and physical systems of nature like simulated annealing [9] and harmony search [6], others were inspired from the success of biological systems in solving problems like genetic algorithm [7,8] and differential evolution (DE) [12], while the most popular algorithms were inspired from swarm intelligence like particle swarm optimization [5], ant colony optimization [3,4], cuckoo search [16], bat algorithm [15], Greywolf optimizer [10] and firefly algorithm [14].

In this paper, we propose a new hybrid Grey Wolf/Bat optimization algorithm (GWOBA). The proposed algorithm is designed to use the best of GWO and BA strategies in order to achieve better overall exploration and exploitation behavior. During the first half of iterations, GWO is used to explore the search space effectively using its high exploration behavior. After GWO is done, BA which has better exploitation capabilities, is initiated using the best set of solutions found by GWO and continues searching for the rest of iterations.

The performance of GWOBA is evaluated using 30 benchmark problems of CEC2017 [1] and is compared to GWO, BA and WOA performance. The results are evaluated using the guidelines provided in CEC2017 and they clearly indicate that GWOBA provides better results in most benchmark problems.

2 Preliminaries

In this section, we review the standard metaheuristics: Grey wolf optimizer (GWO) and bat algorithm. GWO simulates the leadership hierarchy and hunting procedure of grey wolves in nature proposed by Mirjalili et al. in 2014 [10], while Bat algorithm was proposed by Xin-She Yang in 2010 [15], inspired by the echolocation of microbats. Both algorithms have shown superiority over many other metaheuristics over wide range of applications.

2.1 Grey Wolf Optimizer

GWO algorithm is inspired form the hunting mechanism of grey wolves and their social hierarchy. The closest wolves (solutions) to the prey (optimum) are called α wolves. The β and δ wolves are the second and the third best solutions respectively. Their location is denoted in the search space as X_α, X_β and X_δ. The rest of the wolves follow these three wolves as shown in the following equations:

$$X(t+1) = X_p(t) - A.|C.X_p(t) - A.X(t)| \tag{1}$$

where t is the current iteration, A and C are coefficient vectors, X_p is the position vector of the prey, and X indicates the position vector of a grey wolf. The vectors A and C are calculated as follows:

$$A = 2a.r_1 - a,\ C = 2.r_2 \tag{2}$$

where r_1, r_2 are random vectors in $[0, 1]$ and the exploration rate (a) is linearly decreased from 2 to 0 over the course of iterations as shown below:

$$a = 2 - t.\frac{2}{Max_{Iter}} \tag{3}$$

where t is the current iteration and Max_{Iter} is the total number of iterations allowed for the optimization. From Eqs. (2) and (3), the random vector A resides in the interval $[-a, a]$. Exploration and exploitation are guaranteed by the adaptive values of a [10] allowing GWO to transit smoothly between exploration and exploitation.

Supposing that the α, β and δ wolves have better knowledge about the potential location of prey, other agents are obliged to update their positions to follow them.

$$D_\alpha = |C_1.X_\alpha - X|,\ D_\beta = |C_2.X_\beta - X|,\ D_\delta = |C_3.X_\delta - X| \tag{4}$$

where X_α, X_β, and X_δ are the best three solutions at a given iteration.

$$X_1 = X_\alpha - A_1.(D_\alpha),\ X_2 = X_\beta - A_2.(D_\beta),\ X_3 = X_\delta - A_3.(D_\delta) \tag{5}$$

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3} \tag{6}$$

2.2 Bat Algorithm

Bats use echolocation to sense distance and hunt, they emit sound pulses and process the signal of the echo. They can adjust the wavelength, the emission rate and the loudness of the emitted pulses. In the Bat algorithm, bats move randomly with velocity V_i at position X_i with varying wavelength λ and loudness A_0 to search for prey. Virtual bats adjust their position according to the following equations:

$$F_i = F_{min} + (F_{max} - F_{min})\beta \tag{7}$$

$$V_i^t = V_i^{t-1} + (X_i^t - X_*)F_i \tag{8}$$

$$X_i^t = X_i^{t-1} + V_i^t \tag{9}$$

Where β is a random vector in the range [0,1] drawn from uniform distribution. X_* is the current global best location. F_{min} and F_{max} represent the minimum and maximum frequency needed depending on the problem. V_i represents the velocity vector.

Probabilistically a local search is to be performed using a random walk as in the following equation:

$$X_{new} = X_* + \epsilon A^t \tag{10}$$

Where A^t is the average loudness of all bats at this time and ϵ is a random number uniformly drawn from $[-1, 1]$.

The updating of the loudness is performed using the following equation:

$$A_i^{t+1} = \delta A^t \tag{11}$$

Where δ is a constant selected experimentally.

The emission rate r_i controls the application of the local search and is updated using the equation:

$$r_i^{t+1} = r_i^0[1 - exp(-\gamma t)] \tag{12}$$

Where r_i^0 is the initial pulse emission rate and is a constant greater than 0.

3 Proposed Hybrid Grey Wolf-Bat Optimization Algorithm

The hybrid GWOBA algorithm is about mixing the high exploration skills of GWO algorithm with the high exploitative properties of BA. First, GWO algorithm is initialized with a random set of solutions then it iterates to find a better set. After $Max_{Iter}/2$ iterations, only the best two solutions (X_α, X_β) are passed to the bat algorithm (BA) as initial guess to help the algorithm to focus on them. BA then runs for $Max_{Iter}/2$ iterations and returns the best solution which is considered the best solution of the hybrid algorithm.

In order to hybrid GWO and BA algorithms, we needed to make some modifications to both algorithms. The first modification was in GWO algorithm, and the question was: how many leading wolves do we need? By default GWO algorithm makes use of three leading wolves (alpha, beta and delta) which leads the rest of the herd to the prey location. In some problems, the three wolves are located in three different areas within the search space, which slows down the convergence of the algorithm. After many experiments, we decided to use the best two solutions only (alpha and beta) to lead the rest of the solutions to the prey location.

The reason behind this decision is that passing three good solutions to BA might create a divergence, bats are following the best solution and then keep following it with a very low chance to switch this best with another far one. In a three pole function, where one pole is the optimal solution and the other two poles are just local optimas, if GWO has three leaders, having each leader settling down in a pole far away from each other, BA will start searching around them and it might find a best solution around the worst leader and then it will keep searching around the fake best with a very low chance to get out of the pole. This is a typical scenario where GWOBA fails to jump out of a local optima. This modification raised the ability of the algorithm to locate where to search more precisely and helped our algorithm to have significance over the original GWO. Equations (4, 5, 6) are modified to be:

$$D_\alpha = |C_1.X_\alpha - X|,\ D_\beta = |C_2.X_\beta - X| \tag{13}$$

$$X_1 = X_\alpha - A_1.(D_\alpha),\ X_2 = X_\beta - A_2.(D_\beta) \tag{14}$$

$$X(t+1) = \frac{X_1 + X_2}{2} \tag{15}$$

The second modification affects Eq. (3), to enhance the exploration capabilities of the wolves, we forced $|A| > 1$ by decreasing the exploration rate a from two down to one, so Eq. (3) becomes:

$$a = 2 - \frac{t}{Max_{Iter}} \tag{16}$$

where t is the iteration number and Max_{Iter} is the total number of iterations allowed for the optimization. The algorithm describing the hybrid GWOBA algorithm is outlined in Algorithm 1.

Algorithm 1. Algorithm for GWOBA optimizer

```
Input : $n$ Number of agents
        $Max_{Iter}$ Maximum iterations
        $r_i^0$ Initial pulse emission rate
        $A_i$ Loudness
Result: The optimal solution $(X_*)$
Initialize the grey wolf population $X_i$ $(i = 1, 2, ..., n)$ randomly
Initialize $a$, $A$, and $C$
Evaluate the positions of wolves
$X_\alpha$ = The best search agent
$X_\beta$ = The second best search agent
$t = 0$
while $(t < Max_{Iter}/2)$ do
    for each $X_i$ do
        Update $X_i$ position Eq.(15).
    end
    Update $a$, $A$ and $C$ Eqs.(16,2)
    Evaluate the positions of wolves.
    Update $X_\alpha$ and $X_\beta$
    $t = t + 1$
end
Initialize the first two bats $X_1 = X_\alpha$, $X_2 = X_\beta$ and initialize the rest $(n - 2)$ bats $X_i (i = 3, 4, ..., n)$ randomly
Initialize frequencies $f_i$, pulse rates $r_i$ and the loudness $A_i$
Find the best solution based on fitness $X_*$
while $(t < Max_{Iter})$ do
    for each $X_i$ do
        Generate new solution $(X_i^{new})$ by adjusting frequency Eqs.(7 to 9)
        if $rand > r_i$ then
            Generate a local solution $(X_i^{new})$ around the best solution $X_*$ Eq.(10)
        end
        if $rand < A_i$ and fitness $(X_i^{new}) <$ fitness$(X_i)$ then
            Update the position of $X_i$ to $X_i^{new}$
            Reduce $A_i$ Eq.(11)
            Increase $r_i$ Eq.(12)
        end
        Update $X_*$
    end
    $t = t + 1$
end
return $X_*$
```

4 Experimental Results and Discussion

This section summarizes the results from applying the proposed GWOBA on 30 benchmark functions from the new CEC2017 test suit. The next two subsections contain the results and the analysis for the whole set of functions. We use a set of qualitative measures to analyze the results obtained by the methods we apply. The first four metrics (Mean Fitness, Best Fitness, Worst Fitness and Median

Fitness) give a measure of the mean, best, worst, and median expected performance of the algorithms. The fifth measure (Standard Deviation) is adopted to show the precision of each optimizer. The sixth metric (Root-Mean-Square Error (RMSE)) shows how accurate is the optimizer. The last two metrics (T-Test [2,11] and Wilcoxon Rank Sum Test [13]) are used to directly comparing algorithms in pairs and show whether the difference between them is significant or not.

4.1 CEC2017 Benchmark Suite

In this suite [1], benchmark problems were developed with several novel features such as new basic problems, composing test problems by extracting features dimension-wise from several problems, graded level of linkages, rotated trap problems, and so on. All test functions are minimization problems with a shifted global optimum randomly distributed in the range $[-80, 80]$. The suite consists of 30 functions divided into different categories unimodal, simple multimodal, hybrid and composition functions to simulate different types of real life problems.

4.2 Results and Parameter Settings

In order to benchmark our hybrid optimizer, we compared it to its primitive algorithms GWO, BA and to the whale optimization algorithm (WOA) as well. All experiments were done using 20 agents and the standard parameters for all algorithms.

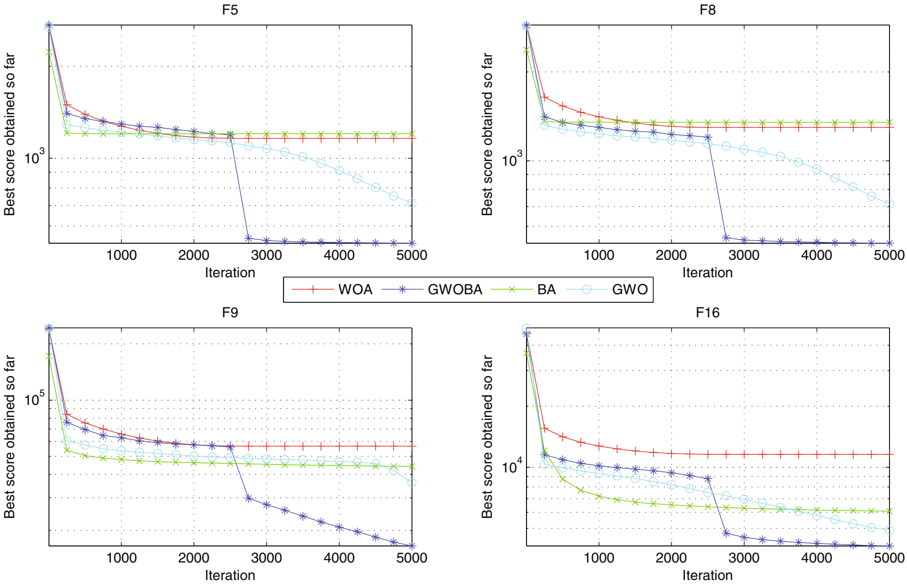


Fig. 1. Mean convergence curves of the compared algorithms (D = 100)

In our experiments, we used 30 minimization problems with four different dimensions: $D = 10, 30, 50, 100$ from CEC2017 benchmark suite. Each optimizer runs 51 runs per problem with maximum function evaluations ($MaxFES = 1000 * D$) (i.e. 10000 function evaluations for 10 dimensions' problem). All problems have the global optimum within the search range $[-100, 100]$. All four optimizers were initialized with uniform random initialization within the search space with a random seed based on time.

In Fig. 1, the four algorithms GWOBA, WOA, GWO and BA are compared using the same dimensions and against the same benchmark functions. We plotted the mean convergence curve for each algorithm along 51 runs for dimension ($D = 100$) and functions ($F5, F8, F9, F16$). As can be noticed, our hybrid algorithm mean convergence curve is diving down when it passes half the maximum number of iterations causing the results to be far better than the other algorithms.

In Table 1, mean error and standard deviation results are tabulated for all 30 benchmark functions ($F1$ through $F30$) along 51 runs for dimension ($D = 100$).

Table 1. Mean error and std. for dimension ($D = 100$

Fun.	Mean				Std.			
	WOA	GWOBA	BA	GWO	WOA	GWOBA	BA	GWO
F01	1.472E+10	**2.580e+04**	4.253E+06	5.985E+10	3.810E+09	**2.226e+04**	3.190E+05	1.271E+10
F02	2.03E+168	**9.65e+108**	5.63E+126	2.79E+138	∞	**6.89e+109**	4.02E+127	1.99E+139
F03	8.674E+05	**2.022e+05**	4.719E+05	2.658E+05	1.450E+05	2.315E+04	1.745E+05	**2.172e+04**
F04	3.217E+03	2.868E+02	**2.817e+02**	5.970E+03	7.679E+02	**3.794e+01**	4.827E+01	1.897E+03
F05	1.162E+03	**5.274e+02**	1.202E+03	7.140E+02	1.008E+02	**4.746e+01**	1.620E+02	8.744E+01
F06	9.264E+01	4.522E+01	7.934E+01	**4.343e+01**	9.672E+00	**3.617e+00**	7.638E+00	4.83E+00
F07	2.834E+03	**1.331e+03**	5.980E+03	1.464E+03	1.414E+02	1.342E+02	1.508E+03	**1.232e+02**
F08	1.298E+03	**5.265e+02**	1.351E+03	7.150E+02	1.193E+02	**6.864e+01**	1.646E+02	8.472E+01
F09	5.665E+04	**1.656e+04**	4.415E+04	3.606E+04	1.583E+04	**2.472e+03**	1.007E+04	1.362E+04
F10	2.451E+04	**1.452e+04**	1.689E+04	1.544E+04	2.088E+03	1.320E+03	**1.150e+03**	3.634E+03
F11	1.282E+05	**1.333e+03**	2.308E+03	6.093E+04	6.306E+04	**1.716e+02**	6.149E+02	1.120E+04
F12	3.119E+09	**3.490e+07**	5.408E+07	1.475E+10	1.020E+09	**1.149e+07**	2.307E+07	8.534E+09
F13	3.088E+07	**5.236e+04**	2.385E+05	1.641E+09	2.149E+07	**2.235e+04**	4.783E+04	1.881E+09
F14	7.826E+06	5.537E+05	**3.948e+05**	6.976E+06	3.195E+06	3.467E+05	**2.031e+05**	3.991E+06
F15	5.848E+06	**4.197e+04**	1.437E+05	2.085E+08	7.563E+06	**1.763e+04**	5.095E+04	3.141E+08
F16	1.158E+04	**4.100e+03**	6.082E+03	4.895E+03	1.617E+03	7.989E+02	1.135E+03	**7.242e+02**
F17	6.806E+03	**3.341e+03**	4.910E+03	3.828E+03	1.536E+03	**5.668e+02**	8.656E+02	1.092E+03
F18	7.209E+06	**7.814e+05**	8.246E+05	5.193E+06	4.270E+06	3.911E+05	**3.515e+05**	2.90E+06
F19	3.903E+07	**1.939e+06**	3.953E+06	2.857E+08	2.872E+07	**8.898e+05**	1.187E+06	4.480E+08
F20	4.466E+03	2.972E+03	4.298E+03	**2.826e+03**	6.726E+02	**5.469e+02**	8.351E+02	7.996E+02
F21	2.067E+03	**7.820e+02**	2.485E+03	9.254E+02	2.607E+02	**6.512e+01**	2.548E+02	9.667E+01
F22	2.579E+04	**1.576e+04**	1.926E+04	1.699E+04	2.100E+03	**1.225e+03**	2.282E+03	2.245E+03
F23	2.697E+03	**1.185e+03**	3.399E+03	1.379E+03	2.426E+02	**7.677e+01**	3.764E+02	1.065E+02
F24	3.716E+03	**1.873e+03**	4.738E+03	1.997E+03	3.616E+02	**1.469e+02**	6.327E+02	1.838E+02
F25	2.552E+03	8.310E+02	**7.933e+02**	4.631E+03	3.216E+02	**4.297e+01**	6.283E+01	9.867E+02
F26	3.124E+04	**1.271e+04**	3.488E+04	1.457E+04	3.640E+03	**1.189e+03**	1.150E+04	1.40E+03
F27	2.737E+03	**1.350e+03**	2.445E+03	1.453E+03	7.615E+02	**1.496e+02**	6.718E+02	1.680E+02
F28	3.464E+03	**6.096e+02**	6.152E+02	6.586E+03	6.093E+02	**4.194e+01**	4.277E+01	1.707E+03
F29	1.298E+04	**5.080e+03**	7.636E+03	6.019E+03	1.676E+03	**5.909e+02**	1.236E+03	7.203E+02
F30	6.088E+08	**1.033e+07**	1.582E+07	1.773E+09	2.941E+08	**3.123e+06**	8.107E+06	1.625E+09

Table 2. Best and worst error for dimension ($D = 100$)

Fun.	Best				Worst			
	WOA	GWOBA	BA	GWO	WOA	GWOBA	BA	GWO
F01	9.395E+09	**8.900e+03**	3.605E+06	2.462E+10	2.542E+10	**1.567e+05**	4.979E+06	8.286E+10
F02	1.94E+142	7.532E+79	**2.027e+63**	2.30E+108	1.01E+170	**4.92e+110**	2.87E+128	1.42E+140
F03	6.257E+05	**1.548e+05**	2.686E+05	2.144E+05	1.338E+06	**2.630e+05**	9.213E+05	3.172E+05
F04	1.797E+03	1.869E+02	**1.847e+02**	2.840E+03	4.976E+03	**3.618e+02**	3.761E+02	1.219E+04
F05	9.415E+02	**4.172e+02**	8.985E+02	5.018E+02	1.386E+03	**6.222e+02**	1.560E+03	1.104E+03
F06	7.553E+01	3.854E+01	6.468E+01	**2.917e+01**	1.183E+02	5.450E+01	9.983E+01	**5.267e+01**
F07	2.485E+03	**1.075e+03**	3.189E+03	1.238E+03	3.107E+03	**1.669e+03**	1.133E+04	1.836E+03
F08	1.117E+03	**3.930e+02**	1.027E+03	5.268E+02	1.650E+03	**6.683e+02**	1.720E+03	9.315E+02
F09	3.478E+04	**1.193e+04**	3.183E+04	1.648E+04	1.088E+05	**2.552e+04**	6.718E+04	5.968E+04
F10	2.018E+04	1.121E+04	1.408E+04	**1.089e+04**	2.887E+04	**1.718e+04**	1.906E+04	2.905E+04
F11	5.518E+04	**9.760e+02**	1.341E+03	4.094E+04	3.581E+05	**1.654e+03**	4.412E+03	9.464E+04
F12	1.343E+09	**1.556e+07**	1.663E+07	3.668E+09	6.470E+09	**5.518e+07**	1.415E+08	4.545E+10
F13	8.183E+06	**1.999e+04**	1.605E+05	1.598E+08	1.036E+08	**1.346e+05**	3.932E+05	1.247E+10
F14	1.855E+06	1.076E+05	**1.074e+05**	1.152E+06	2.031E+07	1.907E+06	**9.684e+05**	1.527E+07
F15	3.192E+05	**1.288e+04**	7.604E+04	1.118E+06	3.197E+07	**8.912e+04**	3.067E+05	1.504E+09
F16	8.931E+03	**2.496e+03**	4.360E+03	3.135E+03	1.571E+04	**5.841e+03**	1.071E+04	6.606E+03
F17	4.676E+03	**1.919e+03**	2.986E+03	2.305E+03	1.270E+04	**4.687e+03**	7.127E+03	6.971E+03
F18	1.953E+06	**2.386e+05**	3.482E+05	1.580E+06	1.860E+07	2.242E+06	**1.874e+06**	1.320E+07
F19	3.510E+06	**3.158e+05**	1.336E+06	4.731E+06	1.311E+08	**4.748e+06**	6.881E+06	2.099E+09
F20	2.486E+03	2.100E+03	2.804E+03	**1.982e+03**	5.703E+03	**4.352e+03**	6.136E+03	5.695E+03
F21	1.513E+03	**6.848e+02**	1.896E+03	7.398E+02	2.713E+03	**9.481e+02**	3.085E+03	1.487E+03
F22	2.074E+04	**1.311e+04**	1.506E+04	1.388E+04	3.072E+04	**1.847e+04**	2.465E+04	2.995E+04
F23	2.145E+03	**9.809e+02**	2.631E+03	1.247E+03	3.184E+03	**1.340e+03**	4.240E+03	1.739E+03
F24	3.163E+03	1.634E+03	3.099E+03	**1.562e+03**	4.710E+03	**2.310e+03**	6.192E+03	2.492E+03
F25	1.945E+03	7.366E+02	**6.503e+02**	2.591E+03	3.598E+03	**9.115e+02**	9.286E+02	7.854E+03
F26	2.367E+04	1.035E+04	**3.143e+02**	1.179E+04	4.015E+04	**1.485e+04**	5.589E+04	1.862E+04
F27	1.726E+03	**9.067e+02**	1.316E+03	1.075E+03	5.046E+03	**1.740e+03**	4.517E+03	1.895E+03
F28	2.432E+03	5.474E+02	**5.450e+02**	3.317E+03	5.215E+03	**7.107e+02**	7.438E+02	1.136E+04
F29	1.013E+04	**3.995e+03**	5.487E+03	4.605E+03	1.677E+04	**6.311e+03**	1.187E+04	7.564E+03
F30	1.689E+08	**3.561e+06**	5.873E+06	1.278E+08	1.838E+09	**1.951e+07**	5.170E+07	5.499E+09

The results show that GWOBA mean error is better than the others for 25 out of 30 functions, even when it fails to achieve the best results, it loses with a very low order of magnitude. Regarding the standard deviation results, GWOBA lost the lead for 6 out of 30 functions, but without order of magnitude.

In Table 2, best and worst error values are tabulated for all 30 benchmark functions ($F1$ through $F30$) along 51 runs for dimension ($D = 100$). The results show that GWOBA found the best solution for 20 out of 30 functions. On the other hand, GWOBA has the lowest worst solutions for 27 out of 30 functions, which means that it might not get the best solution every time but it will not get the worst.

Wilcoxon rank sum test and T-test results with $\alpha = 0.05$ are tabulated in Table 3, the tests are calculated on different dimensions for all four optimizers. Since we are interested in the performance of GWOBA against the other algorithms, we report all the comparisons with GWOBA only. It is clear that

Table 3. Wilcoxon rank sum test and T-test for all dimensions

D	Wilcoxon rank sum test			T-test		
	WOA	BA	GWO	WOA	BA	GWO
10	7.70E-39	3.06E-40	4.25E-07	5.49E-09	1.09E-06	1.17E-02
30	9.81E-44	1.73E-19	3.44E-08	2.00E-13	1.02E-04	2.02E-10
50	1.96E-40	1.08E-15	3.08E-15	3.84E-17	2.75E-04	1.14E-14
100	2.35E-43	1.42E-13	9.51E-26	8.52E-18	3.81E-11	4.23E-19

GWOBA performs better than GWO, BA, and WOA as per T-test results at a significance level $\alpha = 0.05$ for all dimensions.

The last measure to test our hybrid algorithm performance is to sum all wins for the proposed algorithm against GWO, BA, and WOA for different dimensions and for every qualitative measure. Table 4 summarizes the overall performance of GWOBA, showing excellent success rates in mean, std., worst and RMSE measures. The results also show that the algorithm performance is very solid against higher dimensions, while other algorithms fail to get competitive results in the same conditions.

Table 4. Overall success rate of GWOBA for all set of qualitative measures

D	Mean	Std.	Best	Worst	Median	RMSE
10	27	23	13	22	21	26
30	22	27	18	25	20	24
50	22	24	18	21	21	22
100	25	24	20	27	25	25
Success Rate	80%	82%	58%	79%	73%	81%

The mean error and standard deviation results for all dimensions proves that GWOBA continues its great job defeating other algorithms even for the highest dimensions. Having both low mean error and low standard deviation for all dimensions means that GWOBA has the best precision and accuracy compared to the other algorithms. As a result, GWOBA showed excellent performance against its primitive algorithms GWO and BA as well as WOA.

5 Conclusion and Future Work

In this paper, a hybrid GWO/BA algorithm is proposed. The proposed GWOBA algorithm makes use of both GWO and BA strengths in exploration and exploitation. GWOBA is compared to GWO, BA, and WOA on CEC2017 benchmark suite. Results were assessed using a set of performance indicators. Significant improvement was observed in the performance of GWOBA against other methods. GWOBA proved excellent precision and accuracy in different search space terrains with overall best mean value, standard deviation, median, best, worst, and RMSE. The different tests show that GWOBA is very solid regarding higher dimensions as well as lower ones. The algorithm passed two significance tests on four different dimensions. In the near future, we will test GWOBA in other applications to measure its performance in real life problems.

References

1. Awad, N., Ali, M., Liang, J., Qu, B., Suganthan, P.: Problem definitions and evaluation criteria for the cec 2017 special session and competition on single objective real-parameter numerical optimization (2016)
2. Derrac, J., García, S., Molina, D., Herrera, F.: A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol. Comput. **1**(1), 3–18 (2011)
3. Dorigo, M., Birattari, M., Stutzle, T.: Ant colony optimization. IEEE Comput. Intell. Mag. **1**(4), 28–39 (2006)
4. Dorigo, M., Maniezzo, V., Colorni, A.: Ant system: optimization by a colony of cooperating agents. IEEE Trans. Syst. Man Cybern. Part B (Cybern.) **26**(1), 29–41 (1996)
5. Eberhart, R., Kennedy, J.: A new optimizer using particle swarm theory. In: Proceedings of the Sixth International Symposium on Micro Machine and Human Science, MHS 1995, pp. 39–43. IEEE (1995)
6. Geem, Z.W., Kim, J.H., Loganathan, G.: A new heuristic optimization algorithm: harmony search. Simulation **76**(2), 60–68 (2001)
7. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Longman Publishing Co., Inc., Boston (1989)
8. Holland, J.H.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)
9. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., et al.: Optimization by simulated annealing. Science **220**(4598), 671–680 (1983)
10. Mirjalili, S., Mirjalili, S.M., Lewis, A.: Grey wolf optimizer. Adv. Eng. Softw. **69**, 46–61 (2014)
11. Rice, J.: Mathematical Statistics and Data Analysis. Nelson Education (2006)
12. Storn, R., Price, K.: Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. **11**(4), 341–359 (1997)
13. Wilcoxon, F.: Individual comparisons by ranking methods. Biom. Bull. **1**(6), 80–83 (1945)
14. Yang, X.S.: Firefly algorithms for multimodal optimization. In: International Symposium on Stochastic Algorithms, pp. 169–178. Springer (2009)
15. Yang, X.S.: A new metaheuristic bat-inspired algorithm. In: Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), pp. 65–74 (2010)
16. Yang, X.S., Deb, S.: Cuckoo search via lévy flights. In: World Congress on Nature & Biologically Inspired Computing, NaBIC 2009, pp. 210–214. IEEE (2009)

Fractional Order Sliding Mode PID Controller/Observer for Continuous Nonlinear Switched Systems with PSO Parameter Tuning

Ahmad Taher Azar[1,2(✉)] and Fernando E. Serrano[3]

[1] Faculty of computers and information, Benha University, Banha, Egypt
ahmad_t_azar@ieee.org, ahmad.azar@fci.bu.edu.eg
[2] School of Engineering and Applied Sciences, Nile University,
Sheikh Zayed District - Juhayna Square, 6th of October City, Giza 12588, Egypt
[3] Central American Technical University UNITEC, Zona Jacaleapa,
Tegucigalpa, Honduras
serranofer@eclipso.eu

Abstract. In this article a fractional order sliding mode PID controller and observer for the stabilization of continuous nonlinear switched systems is proposed. The design of the controller and observer is done following the separation principle, this means that the observer and controller are designed in a separate fashion, so a hybrid controller is implemented by designing the sliding mode controller part using an integral sliding mode surface along with a $PI^{\lambda}D^{\mu}$ controller part which is the fractional order PID controller that is implemented to stabilizes the system. For the observer part, an integral sliding manifold is implemented, so the error between the measured and estimated variable is reduced to zero when time approaches to infinity. The fractional order PID part is tuned by particle swarm optimization algorithm to derive the λ and μ parameters along with the gain matrices. Finally, the stability of the closed loop system is assured in synchronous switching, this means that the mode in the controller and the system is the same at every switching time, the average dwell time is implemented for this purpose.

Keywords: Fractional order PID · Sliding mode controller
Nonlinear switched systems · Particle swarm optimization

1 Introduction

In this article a fractional order sliding PID controller and observer for the stabilization of continuous switched nonlinear system is proposed. Sliding mode control along with PID control have been implemented for the control of continuous time nonlinear systems. The sliding mode control consists in designing a control law in order to drive the sliding variable to zero in finite time, this fact makes the sliding mode control an efficient control strategy [1–3,8,9,11,15] so

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 13–22, 2018.
https://doi.org/10.1007/978-3-319-74690-6_2

this technique is suitable for the stabilization of several kinds of systems including nonlinear switched systems [4,7]. In the previous two references sliding mode controllers and observers for specific kinds of systems are explained in order to stabilize them. In [7], the controller and observer are designed using a single sliding manifold in order to design the controller and observer, as opposed to the present study in which separated sliding manifolds are implemented to design the observer and controller for the stabilization of nonlinear switched systems. Fractional order sliding mode controller has been implemented recently for the stabilization of different kinds of systems including nonlinear switched systems [10,13] in which due to the flexibility of the fractional order of the controller, this can be tuned by implementing some bio-inspired algorithms to tune the controller parameters and to improve the system performance [11]. In [5], a control strategy for the stabilization of a state delayed system with uncertain perturbation based in sliding mode is shown where the disturbance and state delay of the system are considered. The design of observers for switched systems can be found in [6] considering time delays and unknown outputs and in [13] another observer is proposed for the stabilization of nonlinear synchronous systems. As it is remarked in this section and the rest of this article, only the synchronous case is shown and in the future this study will be extended to the asynchronous case. Finally, it is important to remark that in this study an observer based in [7] is implemented but there are other kinds of observers such as [14] in which a fractional order observer is evinced implemented in a fractional order system, so the results of this study can be extended to other kinds of systems.

Continuous nonlinear switched systems are those kinds of systems in which there are different modes or systems that are switched at different time instants while synchronous continuous nonlinear switched systems are those kinds of systems in which the modes between the controller and the system are the same at any switching times. Taking into account the importance of this kind of systems and their applications in electrical, mechanical, aeronautical systems "etc", it is important to design efficient control strategies considering the complexity of this kind of systems, therefore a fractional order sliding mode PID controller along with a sliding mode observer for the stabilization of this kind of systems in the synchronous case is proposed. The main contribution of this study is that an hybrid control strategy for switched systems is shown considering that the implementation of fractional order PID controllers along with sliding mode controllers is very limited as found in literature. The fractional order sliding mode PID controller is designed by deriving a sliding mode control law selecting a suitable integral sliding mode surface in order to make the sliding variable to reach the sliding surface in finite time, then a fractional order $PI^{\lambda}D^{\mu}$ controller is implemented where the parameters of this controller are tuned by a particle swarm optimization algorithm [12], these parameters are λ and μ along with the fractional PID controller matrix.

The rest of this study is organized as follows. In Sect. 2, Problem Formulation is presented. In Sect. 3, observer design is presented. In Sect. 4, Fractional Order Sliding Mode PID Controller Design is described. Numerical Example and simulation results are presented in Sect. 5. Finally in Sects. 6 and 7, the discussion and conclusion of this study are provided, respectively.

2 Problem Formulation

To derive the controller/observer approach consider the following nonlinear switched system:

$$\begin{aligned} \dot{x}(t) &= A_{\sigma(t)}x(t) + f_{\sigma(t)}(x(t),t) + B_{\sigma(t)}u(t). \\ y(t) &= x(t); \end{aligned} \tag{1}$$

where $x(t) \in \Re^n$, $u(t) \in \Re^m$, $y(t) \in \Re^p$ and $f_{\sigma(t)}(x(t),t)$ is a nonlinear function.

The switching sequence is defined as:

$$\{(i_0,t_0),(i_1,t_1),...,(i_N,t_N)|i_k \in \varGamma\}. \tag{2}$$

where each system is activated in $t \in [t_k, t_{k+1}]$. For each possible value $\sigma(t) = i$ for $i \in \varGamma$, the parameters associated with each subsystem are [7]:

$$\begin{aligned} A_\sigma &\simeq A_i \\ B_\sigma &\simeq B_i \\ f_\sigma(x(t),t) &\simeq f_i(x(t),t) \end{aligned} \tag{3}$$

with this switched system establishment in the following sections the respective controller and observer are designed.

3 Observer Design

The observer is designed first considering an integral sliding manifold and the condition shown in (3). The observer and controller are designed in a separate fashion, this means that in this case the separation principle is implemented. The observer is designed based on the works of [4,6,7] and it is described as:

$$\begin{aligned} \dot{\hat{x}}(t) &= A_i\hat{x}(t) + \hat{f}_i(\hat{x}(t),t) + B_iu_o(t) - L(y(t) - \hat{y}(t)). \\ \hat{y}(t) &= \hat{x}(t) \end{aligned} \tag{4}$$

where $\hat{x}(t) \in \Re^n$, $u_o(t) \in \Re^m$, $\hat{y}(t) \in \Re^p$, $L \in \Re^q$ and $\hat{f}_{\sigma(t)}(\hat{x}(t),t)$ is a nonlinear function. In order to design the observer consider the following sliding manifold [7]:

$$s(t) = K_1\hat{x}(t) + K_2\int_0^t \hat{x}(\tau)d\tau. \tag{5}$$

where K_1 and K_2 are matrices of appropriate dimensions. So,

$$\begin{aligned} \dot{s}(t) &= K_1\dot{\hat{x}}(t) + K_2\hat{x}(t). \\ &= K_1A_i\hat{x}(t) + K_1\hat{f}_i(x,t) + K_1B_iu_o(t) - K_1L(y(t) - \hat{y}(t)) + K_2\hat{x}(t) \end{aligned} \tag{6}$$

and due to,

$$u_o = -B_i^{-1} f_i(x,t) \tag{7}$$

because of:

$$\hat{f}_i(x,t) \simeq f_i(\hat{x},t) \tag{8}$$

Therefore,

$$\dot{s} = K_1 A_i \hat{x}(t) - K_1 L x(t) + K_1 L \hat{x}(t) + K_2 \hat{x}(t) \tag{9}$$

with the previous definitions the following theorem which assured the error convergence to zero when time approaches infinity can be established.

Theorem 1. *The variable $e(t) = y(t) - \hat{y}(t)$ approaches zero as time goes to infinity if the following LMI is solved for L.*

$$\begin{bmatrix} K_1^T K_1 A_i + K_1^T K_1 L + K_1^T K_2 & -K_1^T K_1 L & 0 \\ 0 & 0 & 0 \\ K_2^T K_1 A_i + K_2^T K_1 L + K_2^T K_2 & -K_2^T K_1 L & 0 \end{bmatrix} < 0. \tag{10}$$

Proof. The Lyapunov function (11) is implemented.

$$V(s) = \frac{1}{2} s^T s \tag{11}$$

Therefore, the derivative of $V(s)$ is:

$$\begin{aligned} \dot{V}(s) &= \hat{x}^T(t) K_1^T K_1 A_i \hat{x} - \hat{x}^T(t) K_1^T K_1 L x(t) \\ &+ \hat{x}^T(t) K_1^T K_1 L \hat{x}(t) + \hat{x}^T(t) K_1^T K_2 \hat{x}(t) + \left[\int_0^T \hat{x}(\tau) d\tau\right]^T K_2^T K_1 A_i \hat{x}(t) \\ &- \left[\int_0^T \hat{x}(\tau) d\tau\right]^T K_2^T K_1 L x(t) + \left[\int_0^T \hat{x}(\tau) d\tau\right]^T K_2^T K_1 L \hat{x}(t) \\ &+ \left[\int_0^T \hat{x}(\tau) d\tau\right]^T K_2^T K_2 \hat{x}(t) \end{aligned} \tag{12}$$

making an augmented vector:

$$\hat{x}_a = [\hat{x}, x, \int_0^T \hat{x}(\tau) d\tau]^T. \tag{13}$$

Therefore,

$$\dot{V}(s) < \hat{x}_a \varGamma x_a < 0 \tag{14}$$

then by solving the following LMI the error convergence to zero is assured:

$$\begin{bmatrix} K_1^T K_1 A_i + K_1^T K_1 L + K_1^T K_2 & -K_1^T K_1 L & 0 \\ 0 & 0 & 0 \\ K_2^T K_1 A_i + K_2^T K_1 L + K_2^T K_2 & -K_2^T K_1 L & 0 \end{bmatrix} < 0. \tag{15}$$

so,

$$\dot{V}(s) < 0 \tag{16}$$

With this proof, the error convergence to zero is assured so in the next section the fractional order sliding mode PID controller is derived.

4 Fractional Order Sliding Mode PID Controller Design

In this section, the fractional order sliding mode PID controller is designed. The closed loop system in which the controller, observer and the system are shown in Fig. 1.

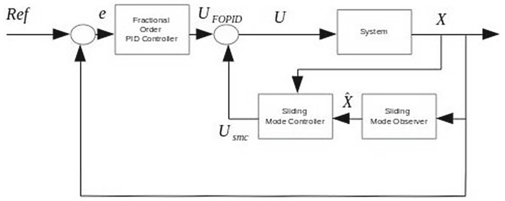


Fig. 1. Closed loop system block diagram

The hybrid control strategy is formed by $U = u_{smc} + u_{FOPID}$ so in the following sections both control laws are obtained along with the particle swarm optimization algorithm to tune the u_{FOPID} controller part [11].

4.1 Sliding Mode Controller Design

Consider the following sliding mode manifold [7]:

$$s(t) = K_1 x(t) + K_2 \int_0^t x(\tau) d\tau \tag{17}$$

$$\dot{s}(t) = K_1 \dot{x}(t) + K_2 x(t) \tag{18}$$

To find the controller for this part, the following theorem is defined as:

Theorem 2. *The sliding mode control law* $u_{smc}(t) = -B_i^{-1} K_1^{-1} K_2 f_i(\hat{x}, t) - B_i^{-1} K_1^{-1} K_2 x(t)$ *stabilizes the nonlinear switched system by solving the following LMI (Linear Matrix inequality) for* K_2*:*

$$\begin{bmatrix} K_1^T K_1 A_i + \frac{\alpha}{2} K_1^T K_1 & \frac{\alpha}{2} K_1^T K_2 \\ K_2^T K_1 A_i + \frac{\alpha}{2} K_2^T K_1 & \frac{\alpha}{2} K_2^T K_2 \end{bmatrix} < 0. \tag{19}$$

Proof. The following Lyapunov function is implemented.

$$V(s) = \frac{1}{2} s^T s \tag{20}$$

taking the first derivative of (20) and using $\dot{s}(t)$ and substituting the control law,

$$u_{smc}(t) = -B_i^{-1}K_1^{-1}K_2 f_i(\hat{x},t) - B_i^{-1}K_1^{-1}K_2 x(t) \tag{21}$$

yields:

$$\begin{aligned}\dot{V}(s) &= x^T(t)K_1^T K_1 A_i x(t)\\ &+ \left[\int_0^t x(\tau)d\tau\right]^T K_2^T K_1 A_i x(t)\end{aligned} \tag{22}$$

so to test the closed loop stability for matched periods (synchronous case), (23) is defined:

$$\begin{aligned}\dot{V}_i(s) + \alpha V_i(s) &< x^T(t)K_1^T K_1 A_i x(t)\\ &+ \left[\int_0^T x(\tau)d\tau\right]^T K_2^T K_1 A_i x(t) + \alpha V_i(s)\end{aligned} \tag{23}$$

then making the vector $x_b = [x, \int_0^T x(\tau)d\tau]^T$, the following condition is met:

$$\dot{V}_i(s) + \alpha V_i(s) < x_b^T \mathit{\Gamma} x_b \tag{24}$$

with a constant α. Therefore with the sliding mode control law (21) and the following LMI, the closed loop stability is achieved.

$$\mathit{\Gamma} = \begin{bmatrix} K_1^T K_1 A_i + \frac{\alpha}{2}K_1^T K_1 & \frac{\alpha}{2}K_1^T K_2 \\ K_2^T K_1 A_i + \frac{\alpha}{2}K_2^T K_1 & \frac{\alpha}{2}K_2^T K_2 \end{bmatrix} < 0. \tag{25}$$

Now in order to consider the closed loop stability of the system in matched periods (synchronous case) starting from (24) and integrating in the range $[t_k, t]$, the following result is obtained:

$$V_i(t) < e^{-\alpha(t-t_k)}V_i(t_k) \tag{26}$$

considering the following condition:

$$V_i(t_k) < \mu V_j(t_k^-) \tag{27}$$

where μ is a constant. Therefore using the following property the closed loop stability for the nonlinear switched system in the synchronous case is needed [7]:

Definition 1. For any $T_2 > T_1 > 0$, let N_σ denotes the number of switching of $\sigma(t)$ over (T_1, T_2) if

$$N_\sigma \leq N_0 + \frac{T_2 - T_1}{T_\sigma} \tag{28}$$

holds for $T_\sigma > 0$ and $N_0 > 0$ where T_σ and N_0 are the average dwell time and the chattering bound respectively. Usually $N_0 = 0$.

Therefore using (26), (27) and Definition 1, the following result is obtained:

$$\begin{aligned} V_i(t) &\leq e^{-\alpha(t-t_k)}\mu V_j(t_{k-1}) \\ &\vdots \\ &\leq e^{(-\alpha+\frac{ln\mu}{T_\sigma})(t-t_0)}V_j(t_0) \end{aligned} \tag{29}$$

so the average dwell time is given by:

$$T_\sigma > \frac{ln\mu}{\alpha} \tag{30}$$

then with this condition the closed loop stability in matched periods or in the asynchronous case is assured.

4.2 Fractional Order PID Controller Design

The fractional order PID controller that is part of the hybrid control law is given by [11]:

$$u_{FOPID} = K_p e(t) + K_i \frac{d^{-\lambda}}{dt^{-\lambda}} e(t) + K_d \frac{d^{\mu}}{dt^{\mu}} e(t) \tag{31}$$

the fractional order PID controller considered in this section is then implemented in the hybrid control law $U = u_{smc} + u_{FOPID}$. The parameters such as the gain matrices and integral and derivative orders are tuned by using a particle swarm optimization algorithm as shown in the following section.

4.3 Parameter Tuning by Particle Swarm Optimization

The parameter tuning is done by using the particle swarm optimization algorithm shown in [12]. The objective function to minimize is the integral square error (ISE) as shown below:

$$F_i(e_i) = \int_0^{t_f} e_i^2(t)dt \tag{32}$$

where $e(t) = x(t) - x_{ref}(t)$ and the parameters to be tuned are the diagonal gain matrices K_p, K_i and K_d along with the orders of the integral and derivative actions λ and μ. The particle position $X(i)$ and velocities $V(i)$ are given by:

$$V(i+1) = V(i) + c1 * rand() * pbest(i) - c2 * rand() * (gbest - X(i)). \tag{33}$$

$$X(i+1) = X(i) + V(i) \tag{34}$$

More details about the particle swarm optimization algorithm can be found in [12].

5 Numerical Example and Simulation Results

In this section, a numerical example to corroborate the theoretical results is shown. The matrices defined in (1) are given by:

$$A_1 = \begin{bmatrix} 0.003 & 0 \\ 0 & 0.004 \end{bmatrix}, A_2 = \begin{bmatrix} 0.0032 & 0 \\ 0 & 0.0043 \end{bmatrix}. \tag{35}$$

$$f_1 = \begin{bmatrix} -2x_1(t)cos(x_2(t)) \\ -3x_1(t)cos(x_2(t)) \end{bmatrix}, f_2 = \begin{bmatrix} -2.1x_1(t)cos(x_2(t)) \\ -3x_1(t)cos(x_2(t)) \end{bmatrix}. \tag{36}$$

and the gain matrices for the observer and controller respectively:

$$L = \begin{bmatrix} 0.008 & 0 \\ 0 & 0.009 \end{bmatrix}, K_2 = \begin{bmatrix} 10.0 & 0 \\ 0 & 10.0 \end{bmatrix}. \tag{37}$$

The number of particles used for the tuning of the fractional order PID parameters is one, the number of parameters are five (three gain matrices, λ and μ) and the number of evolution cycles is five. The obtained integral square error is:

$$F(e) = [0.1, 0.1]; \tag{38}$$

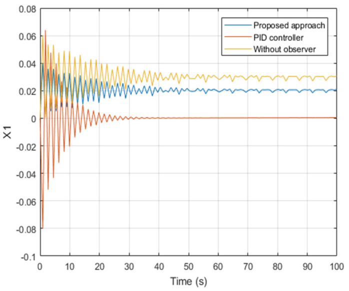


Fig. 2. Variable x_1 for the proposed and compared strategies

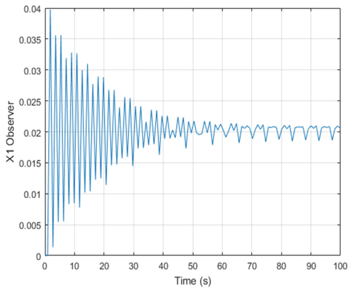


Fig. 3. Variable $\hat{x}_1$ for the proposed and compared strategies

In this example, the proposed strategy is compared with a PID controller and the proposed controller version without observer. As can be noticed in Figs. 2 and 3, the system is stabilized at the operating point $x_{ref} = [0.02, -0.15]$ and it can be concluded that the proposed strategy provides optimal results in comparison with the compared strategy yielded by the accurate observer variable $\hat{x}_1$.

In Fig. 4, the comparison of the control effort for all the strategies is shown, proving that the best result is obtained by the proposed strategy. Finally in Fig. 5, it's shown that the error variable e_1 approaches zero with the proposed strategy in opposition to the compared strategies proving that the best results are obtained by the proposed controller.

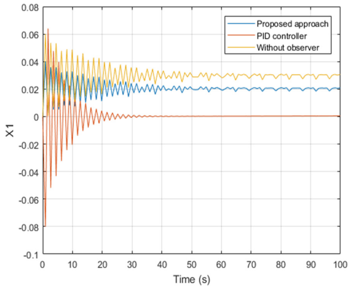


Fig. 4. Control effort U_1 for the proposed and compared strategies

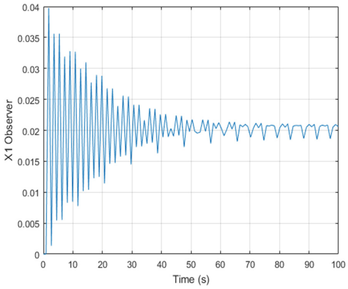


Fig. 5. Error e_1 for the proposed and compared strategies.

6 Discussion

In this article, a novel hybrid control strategy for nonlinear systems and sliding mode observer is proposed. The hybrid control strategy consists of an integer order sliding mode controller along with a fractional order PID controller. As it is corroborated, this hybrid controller strategy improves the system performance in order to follow a reference variable, in this case a step function, proving that the proposed controller is efficient to be implemented in any physical system such as electrical, mechanical or chemical systems just to mention some of them. The sliding mode observer is designed separately using a different sliding manifold than the controller by implementing the separation principle. As it is corroborated in the numerical simulation section that the estimation error is very small so this proves that the observer is designed satisfactorily. The PID fractional controller parameters tuning is done efficiently by using a particle swarm optimization algorithm in order to minimize the integral square error of the system.

7 Conclusion

In this study due to the flexibility of fractional order PID controllers, the stabilization of nonlinear switched systems in synchronous mode is done effectively due to the parameter tuning of the fractional PID controller and because of the capabilities of the sliding mode controller to deal with disturbance and uncertainties. Apart from the control laws design, the stability of the system during matched periods is obtained using the average dwell time in order to meet the stability requirements according to the switching time. A sliding mode observer is implemented in order to estimate the states of the system. The sliding mode controller and observer are designed efficiently by the Lyapunov theorem selecting integral sliding mode manifolds. Finally, by numerical simulation examples, the theoretical results shown in this study are corroborated.

References

1. Azar, A.T., Serrano, F.E.: Adaptive sliding mode control of the furuta pendulum. Advances and Applications in Sliding Mode Control systems, vol. 576, pp. 1–42. Springer International Publishing, Cham (2015)
2. Azar, A.T., Vaidyanathan, S.: Computational Intelligence Applications in Modelling and Control. Studies in Computational Intelligence, vol. 575. Springer, Berlin, Germany (2015)
3. Azar, A.T., Zhu, Q.: Advances and Applications in Sliding Mode Control systems. Studies in Computational Intelligence, vol. 576. Springer, Cham (2015)
4. Gorp, J.V., Defoort, M., Veluvolu, K.C., Djemai, M.: Hybrid sliding mode observer for switched linear systems with unknown inputs. J. Franklin Inst. **351**(7), 3987–4008 (2014)
5. He, Z., Wang, X., Gao, Z., Bai, J.: Sliding mode control based on observer for a class of state-delayed switched systems with uncertain perturbation. Math. Probl. Eng. **2013**, 1–9 (2013). Article ID 614878
6. Lin, J., Gao, Z.: Observers design for switched discrete-time singular time-delay systems with unknown inputs. Nonlinear Anal. Hybrid Syst. **18**(Supplement C), 85–99 (2015)
7. Liu, Y., Niu, Y., Zou, Y.: Non-fragile observer-based sliding mode control for a class of uncertain switched systems. J. Franklin Inst. **351**(2), 952–963 (2014)
8. Meghni, B., Dib, D., Azar, A.T.: A second-order sliding mode and fuzzy logic control to optimal energy management in wind turbine with battery storage. Neural Comput. Appl. **28**(6), 1417–1434 (2017)
9. Mekki, H., Boukhetala, D., Azar, A.T.: Sliding modes for fault tolerant control. In: Azar, A.T., Zhu, Q. (eds.) Advances and Applications in Sliding Mode Control systems, pp. 407–433. Springer International Publishing, Cham (2015)
10. Pashaei, S., Badamchizadeh, M.: A new fractional-order sliding mode controller via a nonlinear disturbance observer for a class of dynamical systems with mismatched disturbances. ISA Trans. **63**(Supplement C), 39–48 (2016)
11. Rahmani, M., Ghanbari, A., Ettefagh, M.M.: Robust adaptive control of a bio-inspired robot manipulator using bat algorithm. Expert Syst. Appl. **56**(Supplement C), 164–176 (2016)
12. Serrano, F., Flores, M.: C++ library for fuzzy type-2 controller design with particle swarm optimization tuning. In: IEEE CONCAPAN 2015, Tegucigalpa, Honduras (2015)
13. Yang, J., Chen, Y., Zhu, F., Yu, K., Bu, X.: Synchronous switching observer for nonlinear switched systems with minimum dwell time constraint. J. Franklin Inst. **352**(11), 4665–4681 (2015)
14. Zhong, F., Li, H., Zhong, S.: State estimation based on fractional order sliding mode observer method for a class of uncertain fractional-order nonlinear systems. Signal Process. **127**(Supplement C), 168–184 (2016)
15. Zhu, Q., Azar, A.T.: Complex System Modelling and Control Through Intelligent Soft Computations. Studies in Fuzziness and Soft Computing, vol. 319. Springer, Berlin, Germany (2015)

Modified Optimal Foraging Algorithm for Parameters Optimization of Support Vector Machine

Gehad Ismail Sayed[1,2(✉)], Mona Soliman[1,2], and Aboul Ella Hassanien[1,2]

[1] Faculty of Computers and Information, Cairo University, Giza, Egypt
DarkSpot_1993@yahoo.com, mona.solyman@fci-cu.edu.eg, aboitcairo@gmail.com
[2] Scientific Research Group in Egypt (SRGE), Giza, Egypt
http://www.egyptscience.net

Abstract. Support Vector Machine (SVM) is one of the widely used algorithms for classification and regression problems. In SVM, penalty parameter C and kernel parameters can have a significant impact on the complexity and performance of SVM. In this paper, an Optimal Foraging Algorithm (OFA) is proposed to optimize the main parameters of SVM and reduce the classification error. Six public benchmark datasets were employed for evaluating the proposed (OFA-SVM). Also, five well-known and recently optimization algorithms are used for evaluation. These algorithms are Artificial Bee Colony (ABC), Genetic Algorithm (GA), Chicken Swarm Optimization (CSO), Particle Swarm Optimization (PSO) and Bat Algorithm (BA). The experimental results show that the proposed OFA-SVM obtained superior results. Also, the results demonstrate the capability of the proposed OFA-SVM to find optimal values of SVM parameters.

Keywords: Optimal Foraging Algorithm · Parameter optimization
Classification · Support vector machine

1 Introduction

Support vector machines (SVMs) are a powerful machine learning technique for both classification and regression problems. SVM is used in many applications in several domains like image analysis, text categorization, and Bio-informatics [1,2]. SVM which originally introduced by Vapnik [3] was based on a learning technique which, following principles from Statistical Learning Theory, and structural risk minimization [4]. Such a learning technique proves high generalization ability in several domains and robustness to high-dimensional data. Such a learning technique aims to achieve a minimization of an upper bound of the generalization error [5]. This is accomplished using penalty parameter as a trade-off between training error and model complexity. With the help of kernel tricks, the SVMs can deal with nonlinear features by mapping to high-dimensional feature space.

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 23–32, 2018.
https://doi.org/10.1007/978-3-319-74690-6_3

The user should specify the values of SVM parameters before the training of SVM. The proper setting of these values has a profound affect on the performance of the obtained classifier. The empirical search for these values through a trial-and-error approach is impractical. Another common approach to optimal parameter selection is creating a parameter grid of different parameter ranges, and to then do an exhaustive grid search over the parameter space to find the best setting of all parameters [6]. Also, such parameters tuning method can result in a large number of evaluations and unacceptably long run times even with moderately high resolution searches.

Recently, heuristic algorithms such as ant colony optimization (ACO), and simulated annealing algorithm (SA), Genetic Algorithm (GA) have been employed to optimize the SVMs parameters for their better global search abilities against numerical optimization methods [7].

In this paper, we develop a new algorithm of SVM parameter tuning using Optimal Foraging Algorithm (OFA), in which The basic OFA was proposed to follow the foraging theory. In this study, OFA is applied to achieve the optimal parameters of SVMs. Experimental results and comparisons demonstrate the effectiveness of the proposed OFA-SVM for parameters optimization of SVMs. The rest of the paper is structured as follows. In Sect. 2, the inspiration of the proposed algorithm is first discussed. Then, the mathematical model is provided. The proposed algorithm is presented and results discussed in Sects. 3 and 4, respectively. Section 5 summarizes the main findings of this paper.

2 Optimal Foraging Algorithm

One of basic animals behavior in the biological life that gains great attention is food foraging. Food provides the animal with energy, But searching for the food requires both energy and time. The animal needs to gain the most benefit (energy) for the lowest cost when he is looking for his food so that it can maximize its fitness. Optimal Forging Theory (OFT) helps the animal to predict the best way to achieve such a goal [8]. If animals can foraging their food in a successful way, their foraging behavior culminates in feeding. The optimal foraging theory addresses the kinds of decisions faced by animals. The main aim of studying the optimal Foraging Theory is the explanation of how animals can use the resource and the dietary patterns of theses animals. The main claim of Optimal Foraging Theory is that: "natural selection favors individuals whose foraging behavior is as energy efficient as possible." In other words, the aim of animal foraging is feeding on the prey with maximizing their net energy intake per unit of foraging time [9].

During last years, Zhang in [10] had proposed a new stochastic search algorithm based on animal foraging behavior. Such proposed algorithm aims to solve global optimization problem and known as Optimal Foraging Algorithm (OFA). The basic OFA was proposed to follow the foraging theory. At foraging time, animals start determining the patch with greatest food plenty in order to start foraging in this positions and its neighborhoods. By the time, animals start consuming food in this patch, so the animals should take the decision to leave this current patch and transfer they foraging for food in other patches. Once the animals start foraging in the new patch, They shall verify first if this prey is beneficial prey or the worthless prey. More and more animals will be attracted if the prey is the beneficial, and profitable prey. Optimal foraging theory have been implemented using different mathematical models [11]. The mathematical model of Optimal Foraging Algorithm (OFA) for solving global optimization problem can be described as follow: Having an objective function $f(x)$ with $F(x^*) = min_{X \in R} f(x)$, $R = (X | x_i^L \leqslant x_i \leqslant x_i^U)$ where X representing d-dimensional state vector $X = [x_1, x_2, ..., x_i, ..., x_d]^T$, $f(X^*)$ represent the optimal objective function with X^* defined the optimal vector or solution, x_i^U, x_i^L are the values of upper and lower bounds respectively.

Forging optimization Algorithm is started by randomly initialize group of P^1 with N individuals using uniform distribution according to the following equation:

$$x_{ji}^1 = x_i^L + rand(0,1) \times (x_i^U - x_i^L) \tag{1}$$

We then evaluate the fitness value F_j^1 for each individual x_j^1 in group P^1, and formulate F_j^1 with $(j = 1, 2, ..., N)$ by ordering F_j^1 from the best to the worse considering the corresponding sequence x_j^1. We then go to iteration process by do the following steps:

- set $t = 1$ for iteration on maximum number of iterations
- set $F_{best} = F_1^t = min(F_1^t, ..., F_N^t)$, $F_N^t = min(F_1^t, ..., F_N^t)$
- check if $(F_j^t = F_{best})$, then update x_{ji}^{t+1} by the following equation:

$$x_{ji}^{t+1} = x_{ji}^t - k \times r_{1ji} \times (x_{Ni}^t - x_{ji}^t) + k \times r_{2ji} \tag{2}$$

  where $r_{1ji} = rand(0,1)$, $r_{2ji} = rand(0,1)$
- check for boundary according to the following:
  if $(x_{ji}^{t+1} > x_i^U)$, then $x_{ji}^{t+1} = 2 \times x_i^U - x_{ji}^{t+1}$
  if $(x_{ji}^{t+1} < x_i^L)$, then $x_{ji}^{t+1} = 2 \times x_i^L - x_{ji}^{t+1}$

- if condition $(F_j^t = F_{best})$ not correct, then according to sorted list F_j^t, choose a value randomly as variable b, then get x_b^t and $F_b^t < F_j^t$.

- update x_{ji}^{t+1} by the following equation:

$$x_{ji}^{t+1} = x_{ji}^t - k \times r_{1ji} \times (x_{bi}^t - x_{ji}^t) + k \times r_{2ji} \tag{3}$$

  where k is defined as $k = \frac{t}{tMax}$ and $tMax$ is the maximum number of iterations. k is linearly decreased through the iterations.
- check for boundary according to the following:
  if $(x_{ji}^{t+1} > x_i^U)$, then $x_{ji}^{t+1} = 2 \times x_i^U - x_{ji}^{t+1}$
  if $(x_{ji}^{t+1} < x_i^L)$, then $x_{ji}^{t+1} = 2 \times x_i^L - x_{ji}^{t+1}$

- Get the objective function value F_j^{t+1} of x_j^{t+1}
- check for the following to determine by the following condition:
  If$(\frac{\lambda_j^{t+1} F_j^{t+1}}{1+\lambda_j^{t+1}(t+1)} < \frac{F_j^t}{t})$, then, $X_j^{t+1} = X_j^{t+1}; F_j^{t+1} = F_j^{t+1}$
  else, $X_j^{t+1} = X_j^t; F_j^{t+1} = F_j^t$
- Formulate F_j^{t+1} with $(j = 1, 2, ..., N)$ by ordering F_j^{t+1} from the best to the worse considering the corresponding sequence x_j^{t+1}.

- check if $(F_1^{t+1} < F_{best})$, then $F_{best} = F_1^{t+1}$; $X_{best} = X_1^{t+1}$

Finally, OFA is terminated by exit the iteration process after getting maximum number of iterations with X_{best} and F_{best}.

3 The Proposed OFA for SVM Parameter Optimization Algorithm

In this section, OFA is used to optimize SVM parameters. Also, the proposed OFA-SVM is applied on Radial Bias Function (RBF) kernel function of the SVM to obtain the optimal solution. The detailed description of OFA-SVM model is as follows:

3.1 Parameters Initialization

At the beginning, the parameters of OFA are initialized. These parameters are number of variables, maximum number of iterations and population size, lower boundary and upper boundary. In this study, two main parameters of RBF kernel function in SVM are optimized. These parameters are penalty parameter C, which controls the classification accuracy and σ parameter, which affecting the partitioning outcome in the feature space [12,13]. The number of variables is set to 2, maximum number of iterations is set to 20, population size is set to 20, lower bound is set for C and σ to 0.01 and upper bound is set for C to 350 and for σ to 100. Then, the initial individuals' positions of OFA are randomly initialized. It should be mentioned that increasing the upper boundary will enlarge the search space. Through this, it will be resulted in slow convergence rate and high computational time. Each position contains values of two variables; C and σ.

3.2 Fitness Function, Positions Updates and Termination Criteria

Through the optimization process, each individual position is evaluated using fitness function Fn_t. In this study, misclassification error rate is used as fitness function (see Eq. (4), where M is the number of misclassified samples and N is total number of samples). Each adopted dataset is randomly divided using K-fold method into training and testing datasets. The training dataset is used to train SVM, while the testing dataset is used to evaluate the an individual position. The optimal position, which represents a solution in the search space is one that obtained the minimum misclassification error rate.

$$Fn_t = minimize(\frac{M}{N}) \tag{4}$$

The updating animals positions of COFA are defined at Eqs. (2) and (3). While, the optimization process terminates when the maximum number of iterations is reached. In our case, the maximum number of iteration is set to 20 iteration.

4 Experimental Results and Discussion

4.1 Datasets Description

In this paper, six well-known datasets obtained from UCI machine learning repository are used to evaluate the performance of the proposed algorithm. Table 1 shows the description of the used datasets [14]. As it can be seen, some of these datasets have missing values. In this study, all these missing values are replaced by the median value of specified feature given the class. The mathematical definition of median method is defined in Eq. (5). $s_{i,j}$ is the missing value for $j-th$ feature of a given $i-th$ class W. For missing categorical values, the most appeared value for a feature given class is replaced with the missing value [15].

$$\bar{s}_{i,j} = median_{i:s_{i,j}\in W_r} s_{i,j} \tag{5}$$

Table 1. Datasets description

ID	Dataset	No. features	No. instances	No. classes	Missing values
D1	Sonar	60	208	2	No
D2	Wisconsin Diagnosis Breast Cancer (WBCD)	32	596	2	No
D3	Glass identification	10	214	6	No
D4	Zoo	18	101	2	No
D5	Cylinder bands	40	512	2	Yes
D6	Diabetes	8	768	2	No

4.2 Results and Discussion

In this paper, the performance of the proposed OFA-SVM is evaluated using six benchmark classification datasets obtained from UCI repository. In addition, the performance of the performance of the proposed algorithm is compared with other optimization algorithms. All the experiments are implemented on the same

Table 2. Parameter settings for PSO, BA, GA, ABC and CSO optimization algorithms.

Algorithm	Parameter	Value
GA	Crossover rate	0.8
	Crossover method	Single crossover
	Mutation rate	0.1
	Mutation method	Bit string
ABC	A number of colony size	10
	A number of food source	5
	A number of limit trials	5
BA	Minimum frequency	0
	Maximum frequency	2
	A	0.5
	r	0.5
CSO	A number of chicken updated	10
	The percent of roosters population size	0.15
	The percent of hens population size	0.7
	The percent of mother hens population size	0.05
PSO	An inertial weight	1
	A inertia weight damping ratio	0.9
	Personal learning coefficient	1.5
	Global learning coefficient	2.0

PC with Core i3 and RAM 2 GB on OS Windows 7. Also, all the obtained results are calculated on average for 10 independent runs. In each run, different training and testing dataset is randomly selected using k-fold cross validation method. In this study, k is set to 10. Each time, one subset is used as the training set and the other $k - 1$ subsets are used as the testing dataset.

Table 3 compares the performance of the proposed OFA-SVM with five other well-known and recent metanephritic optimization algorithms in terms of mean and standard deviation. These algorithms are Artificial Bee Colony (ABC) [16], Genetic Algorithm (GA) [17], Chicken Swarm Optimization (CSO) [18], Particle Swarm Optimization (PSO) [19] and Bat Algorithm (BA) [20]. All these algorithms are used to optimize C and σ of RBF kernel function of SVM. The parameters setting for these algorithms are presented in Table 2. As it can be seen from Table 3, OFA+SVM obtained the minimum misclassification error rate. As it has the minimum mean fitness values. Also, it can be observed from this table that the proposed OFA-SVM has the lowest stability compared with the other algorithms. This high performance of the proposed OFA-SVM is due to the good balancing between exploration and exploitation. However, the low performance of the other algorithms is due to some of these algorithms such as PSO and CSO has many fixed parameters, which need to be tuned first to obtain optimal solution.

Table 3. Statistical results of the proposed OFA-SVM and five other optimization algorithms for six datasets

	OFA		PSO		BA	
	Mean	Std	Mean	Std	Mean	Std
D1	1.364	0	87.997	2.057	12.955	2.305
D2	1.330	1.155	98	0	2	0
D3	23.545	4.876	67.579	16.593	34.616	10.965
D4	6.894	0.722	87.196	7.319	12.805	7.319
D5	6.359	1.09E-15	76.563	1.034	22.787	0.597
D6	21.295	0	71.370	3.401	28.630	3.740
	ABC		CSO		GA	
	Mean	Std	Mean	Std	Mean	Std
D1	14.426	0.243	12.479	2.023	24.916	7.602
D2	2	0	2	0	3.330	2.309
D3	28.122	8.292	26.204	3.892	33.347	15.798
D4	11.824	7.715	12.805	7.319	13.815	6.017
D5	23.050	0.391	23.438	1.408	23.698	0.902
D6	28.630	3.740	28.630	5.230	28.630	5.230

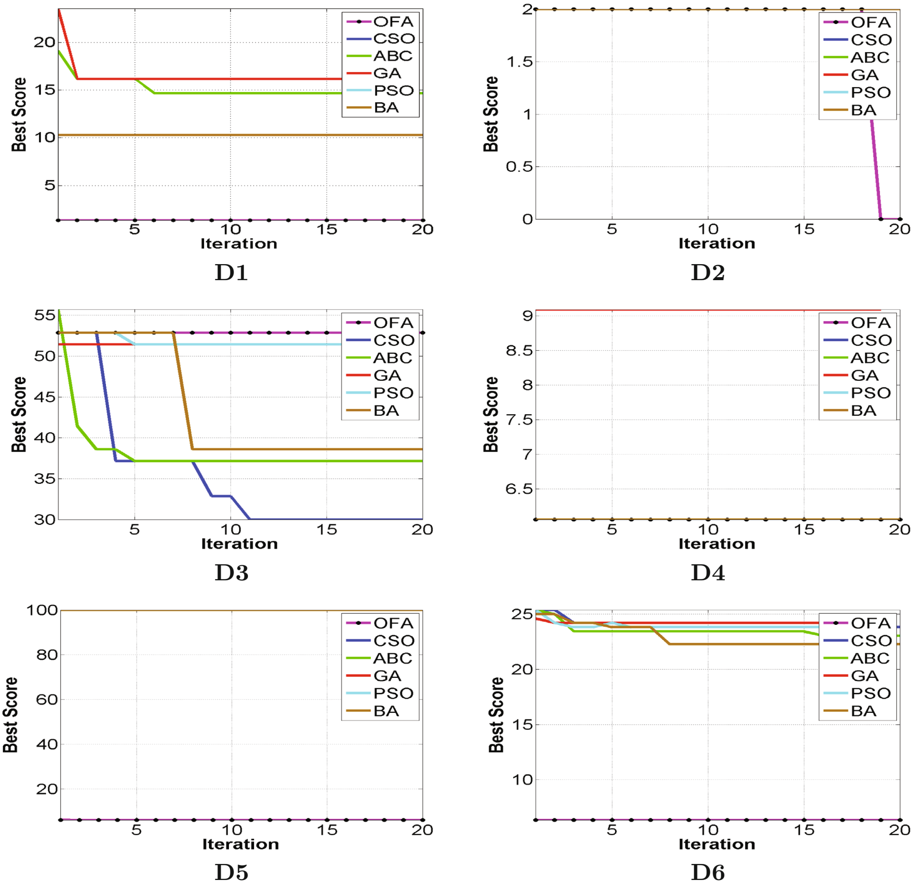
In addition, the parameter k of OFA controls the trade-off between exploration and exploitation phases. As, it linearly decreased through iterations.

Wilcoxon's rank sum test is a nonparametric statistical test [21]. This test is used to prove that the proposed algorithm provides a significant improvement compared to other algorithms. The best values of P when p-values <0.05, which means a sufficient evidence against the null hypothesis. The p-values of Wilcoxon rank sum test is conducted in Table 4. In this table, the classification error of the proposed algorithm is compared with the other algorithms. As it can be seen, most of obtained p-values is less than the significant level 0.05. Also, it can be observed that OFA-SVM outperforms the other optimization algorithms for most benchmark datasets. For further comparison of the proposed OFA-SVM with six

Table 4. OFA vs. five optimization algorithms in terms of p-values of the Wilcoxon ranksum test over six benchmark datasets

	D1	D2	D3	D4	D5	D6
OFA vs. PSO	<0.05	0.50	<0.05	<0.05	<0.05	<0.05
OFA vs. BA	<0.05	0.90	<0.05	0.50	<0.05	0.10
OFA vs. ABC	<0.05	0.84	<0.05	0.53	<0.05	0.22
OFA vs. CSO	<0.05	0.55	0.30	0.12	<0.05	0.40
OFA vs. GA	<0.05	0.60	<0.05	<0.05	<0.05	0.01

benchmark datasets, convergence curves of all the adopted optimization algorithms are analyzed as well. Figure 1 compares the performance of OFA+SVM with ABC+SVM, CSO+SVM, GA+SVM, PSO+SVM and BA+SVM. As it can be observed, OFA+SVM is superior compared with the other algorithms. As, OFA+SVM algorithm converge faster towards the global optima than the other algorithms. Moreover, it can be observed that the proposed OFA for SVM parameters optimization obtains the lowest misclassification error rate. These obtained results are consistent with the obtained results in Table 3. From all the obtained results, it can be concluded that the modified OFA is capable to find the optimal values for C and σ parameters of RBF kernel function in SVM classifier.



D1

D2

D3

D4

D5

D6

Fig. 1. Performance comparison on the six adopted datasets $D1 - D6$

5 Conclusion

In this paper, we proposed an algorithm to optimally tune the penalty parameter C and kernel parameters with modified Optimal Foraging Algorithm (OFA). This work provides new adjusting parameters of SVM approach. The proposed tuning method is considering a simple calculation of implementation with better performances in comparison with the cross-validate method. The proposed tuning parameters had been evaluated over Six well-known benchmark datasets are used for evaluating the proposed (OFA-SVM). Also, the performance of the proposed algorithm (OFA-SVM)is compared with five well-known swarm optimization algorithms. From all the obtained results, it can be concluded that the modified OFA is capable of finding the optimal values for C and σ parameters of RBF kernel function in SVM classifier.

References

1. Sayed, G., Hassanien, A.: Moth-flame swarm optimization with neutrosophic sets for automatic mitosis detection in breast cancer histology images. Appl. Intell. **47**(2), 397–408 (2017)
2. Sayed, G., Hassanien, A., Ibrahim, M.: Particle swarm optimization and k-means algorithm for chromosomes extraction from metaphase images. In: 3rd International Conference on Advanced Intelligent Systems and Informatics (AISI 2017), Cairo, Egypt, pp. 331–341. Springer (2017)
3. Chapelle, O., Vapnik, V., Bousquet, O., Mukherjee, S.: Choosing multiple parameters for support vector machines. Mach. Learn. **46**(1–3), 131–159 (2002)
4. Lin, S., Ying, K., Chen, S., Lee, Z.: Evolutionary tuning of SVM parameter values in multiclass problems. Neurocomputing **71**(4), 3326–3334 (2008)
5. Luo, Z., Zhang, W., Li, Y., Xiang, M.: SVM parameters tuning with quantum particles swarm optimization. In: IEEE conference on Cybernetics and Intelligent Systems, Chengdu, China, pp. 183–187 (2008)
6. Friedrichs, F., Igel, C.: Evolutionary tuning of multiple SVM parameters. Neurocomputing **64**, 107–117 (2005)
7. Zhang, L., Wang, J.: Optimizing parameters of support vector machines using team-search-based particle swarm optimization. Eng. Comput. **32**(5), 1194–1213 (2015)
8. Sinervo, B.: Optimal foraging theory: constraints and cognitive processes. In: Behavioral Ecology, chap. 6, pp. 105–130. University of California, Santa Cruz (1997)
9. Krebs, J.R., Erichsen, J.T., Webber, M.I.: Optimal prey selection in the great tits (parus major). Anim. Behav. **25**(1), 30–38 (1977)
10. Zhu, G., Zhang, W.: Optimal for aging algorithm for global optimization. Appl. Soft Comput. **51**, 294–313 (2016)
11. Pyke, G.H., Pulliam, H.R., Charnov, E.L.: Optimal foraging: a selective review of theory and tests. Q. Rev. Biol. **52**(2), 37–154 (1977)
12. Tharwat, A., Gabel, T., Hassanien, A.: Parameter optimization of support vector machine using dragonfly algorithm. In: Proceedings of the International Conference on Advanced Intelligent Systems and Informatics, Egypt, pp. 309–319 (2017)

13. Sayed, G., Hassanien, A., Kim, T.: Interphase cells removal from metaphase chromosome images based on meta-heuristic grey wolf optimizer. In: 11th International Computer Engineering Conference (ICENCO), Cairo, Egypt, pp. 261–266. IEEE (2015)
14. Bache, K., Lichman, M.: UCI Machine Learning Repository. http://archive.ics.uci.edu/ml
15. Sayed, G., Soliman, M., Hassanien, A.: Bio-inspired swarm techniques for thermogram breast cancer detection. In: Medical Imaging in Clinical Applications. Studies in Computational Intelligence, vol. 651, pp. 487–506. Springer, Cham (2016)
16. Wang, H., Zhang, H., Cang, S., Liao, W., Zhu, F.: Parameters optimization of classifier and feature selection based on improved artificial bee colony algorithm. In: Proceedings of the International Conference on Advanced Mechatronic Systems, Melbourne, Australia, pp. 242–247 (2016)
17. Huang, C., Wang, C.: A GA-based feature selection and parameters optimization for support vector machines. Expert Syst. Appl. **31**(2), 231–240 (2006)
18. Taie, S., Ghonaim, W.: Title CSO-based algorithm with support vector machine for brain tumar's disease diagnosis. In: IEEE International Conference on Persasive Computing and Communications Workshops, Kona, USA, pp. 183–187 (2017)
19. Lin, S., Ying, K., Chen, S., Lee, Z.: Particle swarm optimization for parameter determination and feature selection of support vector machines. Expert Syst. Appl. **35**(4), 1817–1824 (2008)
20. Taie, S., Ghonaim, W.: Adjusted bat algorithm for tuning of support vector machine parameters. In: IEEE Congress on Evolutionary Computation (CEC), Vancouver, Canada, pp. 2225–2232 (2016)
21. Sayed, G., Hassanien, A., Azar, A.: Feature selection via a novel chaotic crow search algorithm. In: Neural Computing and Applications, pp. 1–18. Springer, London (2017)

Pareto Based Bat Algorithm for Multi Objectives Multiple Constraints Optimization in GMPLS Networks

Mohsin Masood[1](✉), Mohamed Mostafa Fouad[2], and Ivan Glesk[1]

[1] Electronics and Electrical Engineering Department, University of Strathclyde, Glasgow, UK
{mohsin.masood,ivan.glesk}@strath.ac.uk
[2] Arab Academy for Science, Technology and Maritime Transport, Cairo, Egypt
mohamed_mostafa@aast.edu

Abstract. Modern communication networks offer advance and diverse applications, which require huge usage of network resources while providing quality of services to the users. Advance communication is based on multiple switched networks that cannot be handle by traditional IP (internet protocol) networks. GMPLS (Generalized multiprotocol label switched) networks, an advance version of MPLS (multiprotocol label switched networks), are introduced for multiple switched networks. Traffic engineering in GMPLS networks ensures traffic movement on multiple paths. Optimal path(s) computation can be dependent on multiple objectives with multiple constraints. From optimization prospective, it is an NP (non-deterministic polynomial-time) hard optimization problem, to compute optimal paths based on multiple objectives having multiple constraints. The paper proposed a metaheuristic Pareto based Bat algorithm, which uses two objective functions; routing costs and load balancing costs to compute the optimal path(s) as an optimal solution for traffic engineering in MPLS/GMPLS networks. The proposed algorithm has implemented on different number of nodes in MPLS/GMPLS networks, to analysis the algorithm performance.

Keywords: Bat algorithm · GMPLS networks · Optimization
Particle swarm optimization · Routing protocols · Traffic engineering

1 Introduction

Advance telecommunication applications require a massive movement of data flow in the network, which causes various network problems such as congestion, packet delays, high utilization of network resources and bandwidth use [1]. To address these challenges, traffic engineering concept was introduced in the networks. Traffic engineering (TE), is used to optimize the network performance by ensuring massive data flow in the network with minimum utilization of network resources and with performance efficiency. TE can be applied to any range (from local area to wide area) of multiple switched networks. Recently, multiple path traffic engineering has been introduced as appealing approach to handle diverse applications with increased network performance [2]. Multiple path routing is the technique of traffic management, which balances large amount of traffic

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 33–41, 2018.
https://doi.org/10.1007/978-3-319-74690-6_4

into multiple routes. It shows significant results compare to traditional routing techniques, which relies on forwarding traffic over shortest path routes. Multipath traffic engineering optimizes network utilization and address various network problems effectively such as packet loss, congestion and link loads. Multipath routing traffic engineering requires algorithms which can compute optimal routes, having multiple objectives and constraints [3, 4]. In networking, it is known as multi-objective multiple constrained (MCOP) based optimization problem, which is an NP hard. This paper provides a meta-heuristic pareto based bat algorithm, which will provide optimal solutions as paths for MCOP in communication networks.

Traditional IP networks has various limitation while using traffic engineering, which affects traffic engineering performance. Therefore, to improve network capabilities, multiprotocol label switched (MPLS) networks are introduced, which are based on label switched network. Furthermore, Generalized multiprotocol label switched (GMPLS) network is introduced, which is the extended version of MPLS networks. GMPLS networks provide the set of protocols which enable forwarding of traffic over multiple switched networks such as packet, time, wavelength and fiber switching networks [5, 6].

The proposed algorithm considers two objective functions; routing costs and load balancing costs with constraints and the task is to find the optimal paths (as solutions) in MPLS/GMPLS networks.

2 MPLS/GMPLS Networks

MPLS/GMPLS uses labels over the packets and forward them in the network from source to destination routers. Routing protocols play an important role for label switching and forwarding of packets in MPLS/GMPLS networks [6]. In MPLS/GMPLS domain, a virtual connection is established known as label switched path (LSP) for forwarding user data. The establishment of the label switched paths (LSP) is done with the help of interior gateway routing protocols such as open shortest path first (OSPF) and intermediate system-to-intermediate system (IS-IS) protocols [7]. When the packet arrives from the source, the router connected to source site label the packet and forward to its next router towards the destination. Each intermediate router in the network lookup the label and forward the packet to the next routers in the network, unless the packet reaches to the router at destination site. The routers at source and destination site, are known as label edge routers (LER) while the routers, used for forwarding labelled packets, are known as label switched routers (LSR). Router connected to source site, which receives traffic request and take the initiative for label switched path (LSP) is known as ingress router. While the label edge router (LER) which is at destination side is known as egress router. Label switched path (LSP) develops between ingress and egress routers in MPLS/GMPLS domain. Once the path or label switched path (LSP) has established, then the user data will forward from source to destination through label switched routers (LSRs) in the network. This label switching approach of MPLS/GMPLS networks enhances network performance with minimum utilization of network resources compare to IP networks, where each router must look up the list of IP addresses. Most of the service providers prefer GMPLS based routers for modern applications [7, 8].

3 Problem Evaluation

To provide the effective traffic engineering in MPLS/GMPLS network and for handling massive amount of traffic flow, the techniques must be used which can enhance network performance and provide optimal solutions. In MPLS/GMPLS networks, ingress receives a number of traffic requests, and the task is to find the number of optimal routes while considering multiple objectives and constraints. An algorithm can offer optimal paths as solutions for the given scenario. In the paper, we proposed pareto based bat algorithm, while considering two objective functions; routing costs and load balancing costs. The proposed algorithm will provide optimal solutions as paths having minimum routing costs and load balancing costs. The algorithm will be implemented on different number of nodes in MPLS/GMPLS networks for analyzing network performance.

In the paper, we used notation for MPLS/GMPLS networks as graph(G). The network/graph(G) is consist of number of routers and links, which are represented as; for routers set, vertices(V) is used and for links set, edges(L) is used. The graph with number of vertices and edges can be represent as $G = (V, L)$. The set of vertices (V) in the network is $\mathrm{V} = \left\{v_1, v_2, v_3, \ldots, v_n\right\}$ and links set is $L = \left\{l_1, l_2, l_3, \ldots, l_n\right\}$. The objective functions are explained as follow.

3.1 Total Routing Costs Objective Function

Service providers use specific link cost for per unit of data flow in MPLS/GMPLS networks, which is described as follow [9, 10]:

$$R_{cost} = \sum T_{links} I_{traffic} \tag{1}$$

Where, R_{cost} represents the routing cost for a path. While T_{links} represents the connected links and $I_{traffic}$ is the i^{th} traffic over the path. The total routing costs objective function is mathematically described as follow [9, 10]:

$$1^{\mathrm{st}}\ \text{Objective Function} = \sum traffic \in T_{traffic} \sum R_{cost} \tag{2}$$

Where, *traffic* is member of all traffics set($T_{Traffic}$).

3.2 Total Load Balancing Costs Objective Function

The second objective function is to distribute the traffic evenly over multiple links, which is dependent on load balancing costs. Load balancing costs function consist of two parameters, known as link utilization(L_u) and link capacity(L_c). The load balancing function can be described as follow [9, 10]:

$$Load_{balancing} = \text{link utilization}\left(L_u\right)/\text{link capacity}\left(L_c\right) \tag{3}$$

In our experiments, the task of the proposed algorithm is to minimize the load balancing function. The mathematical expression for the total load balancing costs is given as follow [9, 10]:

$$2^{\mathrm{nd}}\ \text{Objective Function} = \min\left(\sum Load_{balancing}\right) \tag{4}$$

4 Proposed Algorithm

We proposed a metaheuristic algorithm to address the optimization problem in traffic engineering for MPLS/GMPLS networks.

4.1 Pareto Based Bat Algorithm (PBA)

Bat algorithm is a mathematic bio-inspired technique introduced by Yang in 2010 [11], which is used for solving optimization problems in different applications. Bat algorithm is inspired by the bat technique for searching its prey in searching area. While searching for its prey, each bat periodically evaluates its searching as updated solutions with the given fitness function. The searching nature of bats dependent on echolocation parameters known as loudness(L_d) and pulse-rate(P_r). When the bat approaches towards its prey, the loudness(L_d) decreases while pulse-rate(P_r) increases [11, 12]. In our paper, we modelled bat algorithm as Pareto based model, in which each bat will search for optimal solutions as minimum routing costs and minimum load balancing costs paths in n-dimension searching space. In bat algorithm, each (i^{th}) bat is used as a candidate of searching optimal solution, where it updates its position$\left(x_i^{ite}\right)$ and velocity$\left(v_i^{ite}\right)$ in n-dimension searching space during each iteration, which is given as follow [11–13]:

$$freq_i = freq_{min} + \beta\left(freq_{max} + freq_{min}\right) \tag{5}$$

$$v_i^{ite} = v_i^{ite-1} + freq_i(x_i - x^{globalbest}) \tag{6}$$

$$x_i^{ite} = x_i + v_i^{ite} \tag{7}$$

Where *ite* represents the iterations used in the algorithm. $freq_i$ represents the initial frequency while$freq_{max}$ and$freq_{min}$ are the maximum and minimum frequencies, respectively. β is the random number within the range of 0 and 1. $x^{globalbest}$ is global best position of the i^{th} bat. The global best position$\left(x^{globalbest}\right)$ is accomplished by comparing all given solutions of n bats. Each bat, after updating its velocity$\left(v_i^{ite}\right)$ and position$\left(x_i^{ite}\right)$ takes a random walk for searching to achieve its local best solution based on the condition; *if rand > pulse-rate (P_r),* based on following [11–13]:

$$x_{i,best\text{-}local}^{ite} = x_i + \varepsilon < L_{d,Aveg} \tag{8}$$

where, $x^{ite}_{i,best\text{-}local}$ is used for local best position. ε is a random number, $\varepsilon \in [-1, 1]$. $L_{d,Aveg}$ represents the average loudness of the bats. During each iteration, bat updates its loudness(L_d) and pulse rate(P_r) value. If the bat is approaching to its optimal solution then the loudness(L_d) level will decrease while pulse-rate(P_r) level will increase, as given by following equations [11–13]:

$$L^{ite+1}_{d,i} \;=\; \alpha\, L^{ite}_{d,i} \tag{9}$$

$$P^{ite}_{r,i} = P_{r,i}[1 - e^{-\gamma t}] \tag{10}$$

Where, α and γ are constant values, set from the interval of [0, 1]. The pseudo code of the proposed pareto based bat algorithm is given in Algorithm 1.

Algorithm 1. Pseudo code of Pareto based Bat Algorithm (PBA)

```
Routing costs objective function $f_{x,\,routing} = [x_{r,1}, x_{r,2}, x_{r,3}, \ldots\ x_{r,n}]$
Load balancing costs objective function $f_{x,\,load} = [x_{l,1}, x_{l,2}, x_{l,3}, \ldots\ x_{l,n}]$
Remove the links and routers from the matrix, after applying the constraints associated to routing costs and load balancing costs functions
Initialize number of bats population
    At initial pulse-rate($P_r$) and initial loudness($L_d$), initialize pulse frequency($freq_i$)
While (iterations < total number of iterations for routing costs function)
        Update frequency($freq_i$) by adjusting maximum($freq_{max}$) and minimum frequency($freq_{min}$)
        Update bats position($x_i^{ite}$) and velocities($v_i^{ite}$) in the network (matrix)
        Apply the routing costs function constraints.
        Generate local best position of each $i^{th}$ bat
    if (rand < Pulse-rate($P_r$))
            Generate local optimal solution as a path having minimum routing costs
    end if
        Generate random solutions (paths) in the matrix randomly
    if (rand < $L_d$ & Present routing costs < Previous routing costs)
            Accept the new updated solution as optimal path
            Increase Pulse-rate($P_r$) and decrease Loudness($L_d$)
            Find the global best position($x^{globalbest}$) of the $i^{th}$ bat having optimal solution
    end if
end While
Store the optimal solutions as paths having minimum routing costs
While (iterations < total number of iterations for Load balancing costs function)
        Update ($freq_i$) by adjusting $freq_{max}$ and $freq_{min}$
        Update bats position($x_i^{ite}$) and velocities($v_i^{ite}$) in the network (matrix)
        Apply the load balancing costs function constraints.
        Generate local best position of each $i^{th}$ bat
    if (rand < Pulse-rate($P_r$))
            Generate local optimal solution as a path having minimum load balancing costs
    end if
        Generate random solutions (paths) in the matrix randomly
    if (rand < $L_d$ & Present load balancing costs < Previous load balancing costs)
            Accept the new updated solution as optimal path
            Increase Pulse-rate($P_r$) and decrease Loudness($L_d$)
            Find the global best position($x^{globalbest}$) of the $i^{th}$ bat having optimal solution
            Store the optimal solutions as paths having minimum load balancing costs
    end if
end While
Store the optimal solutions as paths having minimum routing costs
Generate Pareto archive of paths with minimum routing and load balancing costs
```

5 Experimental Setup

Throughout the experiments the algorithm had been implemented as pareto based bat algorithm using MATLAB tool. For analyzing performance analysis of the proposed algorithm, it was implemented over various scales of nodes in MPLS/GMPLS networks such as 80, 90 and 100 nodes, as presented in Figs. 1, 2 and 3 respectively. Furthermore, the proposed algorithm has been modified through changing its parameters and then divide them into five cases, entitled as PBA-1 (Pareto based bat algorithm), PBA-2, PBA-3, PBA-4 and PBA-5. In each PBA case, we changed the maximum loudness value ($L_{d,\ max}$) and minimum loudness value ($L_{d,\ min}$), which updates the loudness(L_d) value during iteration. In PBA-1; $L_{d,\ max} = 5$, in PBA-2; $L_{d,\ max} = 12$, in PBA-3; $L_{d,\ max} = 18$, in PBA-4; $L_{d,\ max} = 24$ and in PBA-5; $L_{d,\ max} = 30$, while $L_{d,\ min} = 0$ for all PBA cases. Pareto based optimal solutions of two objective functions simulated results are shown in Figs. 1, 2 and 3, with Pareto frontiers. The paper highlighted the non-dominated solution of both objective functions with different signs and connect then with lines to draw a Pareto front for each case.

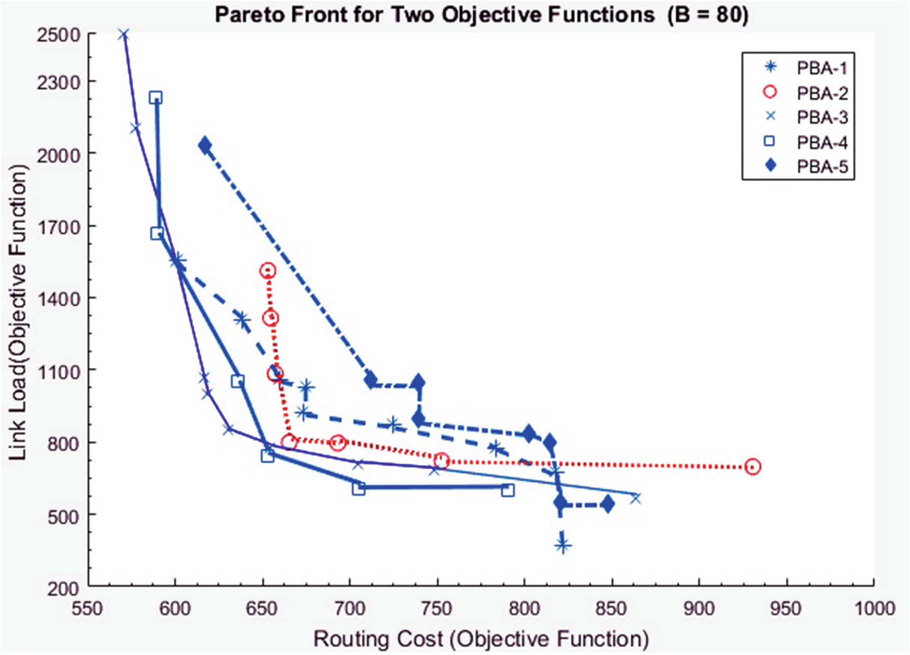


Fig. 1. Pareto front of routing costs and load balancing costs function for nodes (B) = 80

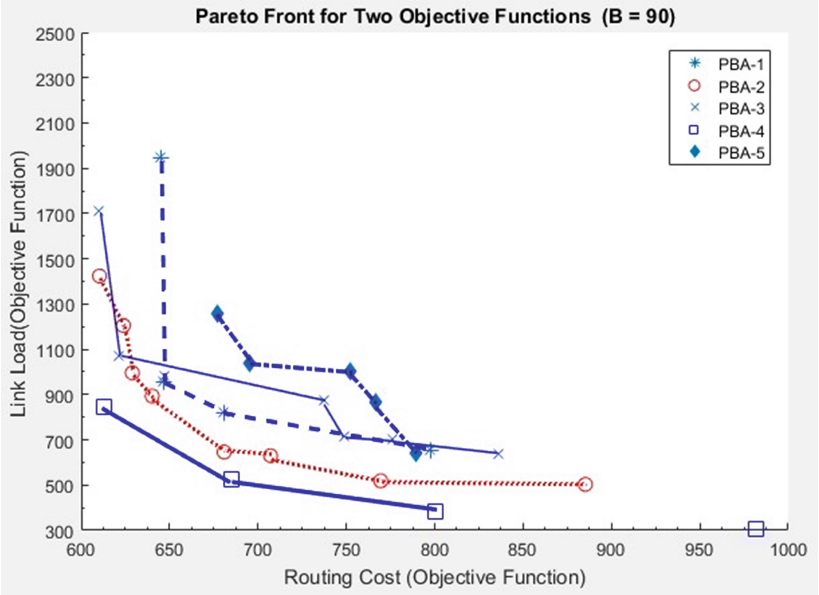


Fig. 2. Pareto front of routing costs and load balancing costs function for nodes (B) = 90

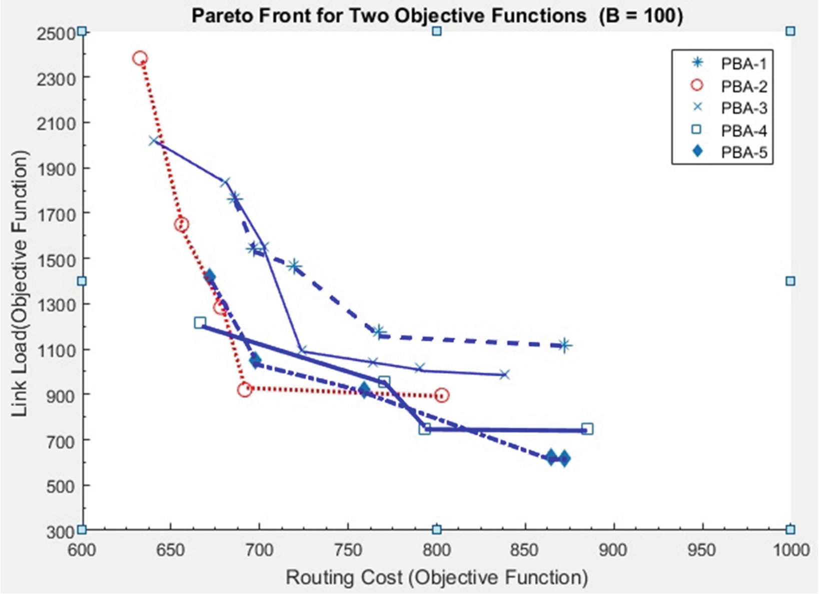


Fig. 3. Pareto front of routing costs and load balancing costs function for nodes (B) = 100

6 Result Analysis

The figures represent the optimal solutions (paths) for two objective functions, where each solution represents the minimum routing costs and load balancing costs. For example, in Fig. 1, for PBA-1 case in 80 nodes network, the Pareto curve shows the optimal solutions with highlighted points which are connected lines. It is also noticed that when routing costs increase, the load balancing costs decreases and vice versa. Routing costs and load balancing costs are minimum/optimal values (as shown in Pareto front) in 80 nodes network compare to 90 and 100 nodes networks for all PBA scenarios. Similarly, 90 nodes network has better results compare to 100 nodes networks. These findings are same for PBA-2, PBA-3, PBA-4 and PBA-5 for 80, 90 and 100 MPLS/GMPLS nodes networks, as shown in all figures.

For comparative analysis, the proposed Pareto based BAT algorithm (PBA) is compared with particle swarm optimization algorithm (PSO). Each algorithm is implemented on 100 nodes GMPLS network. The parameters used for comparison are: minimum routing costs, minimum load balancing costs, mean values and standard deviation. Both algorithms run for 100 times to collect data and then analyze with mentioned parameters, which is presented in Table 1. The results in Table 1 show that proposed bat algorithm (PBA) has minimum or optimum values for both routing costs and load balancing costs function, in addition to reduction other measuring's parameters. For example, PBA algorithm has minimum routing costs value of 462 compare to PSO routing costs value of 1169, which means that PBA algorithm achieved optimum value compared to PSO algorithm. Similarly, for mean values and standard deviation values; PBA algorithm achieved minimum (optimum) values compare to PSO algorithm obtained values, which shows that PBA algorithm obtains optimum values as a mean with a small standard deviation from the mean. This may have related to the adjustment of the frequency of the bat based on how far is the object.

Table 1. Comparative study table between proposed Pareto BAT (PBA) and PSO

	100 Nodes MPLS/GMPLS network					
	Minimum routing costs	Mean (Routing costs)	Standard deviation (Routing costs)	Minimum load balancing costs	Mean (Load balancing costs)	Standard deviation (Load balancing costs)
Proposed PBA	463	865	150.29	87	150	100
PSO	1169	177	269	101	260	125

7 Conclusion

The paper has presented the metaheuristic based algorithm as a solution for multiple constrained based multi-objective optimization (MCOP) problem for traffic engineering in MPLS/GMPLS networks. The proposed algorithm (with its presented pseudo code) is implemented on different number of nodes in MPLS/GMPLS network with various algorithm cases such as PBA-1, PBA-2, PBA-3, PBA- 4 and PBA-5. The algorithm provides optimal solutions with Pareto front for minimum routing costs and load balancing costs. We also found that the routing costs increases when load balancing costs decreases and vice versa. Furthermore, the optimal solutions in the form of Pareto front have minimum routing costs and load balancing costs in small networks compare to large MPLS/GMPLS networks.

References

1. Walrand, J., Varaiya, P.: High-Performance Communication Networks. Elsevier Science, San Francisco (1999)
2. Mazandu, G.: Traffic Engineering Using Multipath Routing Approaches (2007)
3. Liu, H., Zhang, X., Wang, D., Xu, G.: An algorithm for end-to-end performance analysis of network based on traffic engineering. J. Electron. (China) **20**(4), 293–298 (2003)
4. Girão-Silva, R., Craveirinha, J., Clímaco, J., Captivo, M.: Multiobjective routing in multiservice MPLS networks with traffic splitting — a network flow approach. J. Syst. Sci. Syst. Eng. **24**(4), 389–432 (2015)
5. Ramadža, I., Ožegović, J., Pekić, V.: Network performance monitoring within MPLS traffic engineering enabled networks. In: 2015 23rd International Conference on Software, Telecommunications and Computer Networks (SoftCOM). IEEE (2015)
6. Lv, M., Ji, W.: Research on GMPLS traffic engineering mechanism. In: IEEE 13th International Conference on Communication Technology (ICCT). IEEE (2011)
7. Masood, M., Abuhelala, M., Glesk, I.: A comprehensive study of routing protocols performance with topological changes in standard networks. Int. J. Electron. Electr. Comput. Syst. **5**(8), 31–40 (2016)
8. Farrel, A., Bryskin, I.: GMPLS. Elsevier/Morgan Kaufman, San Francisco (2006)
9. El-Alfy, E., Mujahid, S., Selim, S.: A Pareto-based hybrid multiobjective evolutionary approach for constrained multipath traffic engineering optimization in MPLS/GMPLS networks. J. Netw. Comput. Appl. **36**(4), 1196–1207 (2013)
10. Erbas, S.C., Erbas, C.: A multiobjective off-line routing model for MPLS networks. In: Proceedings of the 18th International Teletraffic Congress (2003)
11. Yang, X.S.: A new metaheuristic Bat-inspired algorithm. In: González, J.R., Pelta, D.A., Cruz, C., Terrazas, G., Krasnogor, N. (eds.) Nature Inspired Cooperative Strategies for Optimization (NICSO 2010). Studies in Computational Intelligence, vol. 284. Springer, Heidelberg (2010)
12. Malakooti, B., Kim, H., Sheikh, S.: Bat intelligence search with application to multi-objective multiprocessor scheduling optimization. Int. J. Adv. Manuf. Technol. **60**(9–12), 1071–1086 (2011)
13. Castelo Damasceno, N., Gabriel Filho, O.: PI controller optimization for a heat exchanger through metaheuristic Bat algorithm, particle swarm optimization, flower pollination algorithm and Cuckoo search algorithm. IEEE Lat. Am. Trans. **15**(9), 1801–1807 (2017)

Fish Image Segmentation Using Salp Swarm Algorithm

Abdelhameed Ibrahim[1,4](✉), Ali Ahmed[2],
Sherif Hussein[1], and Aboul Ella Hassanien[3,4]

[1] Faculty of Engineering, Mansoura University, Mansoura, Egypt
afai79@mans.edu.eg
[2] Faculty of Computers and Information, Menoufia University,
Shibin El Kom, Egypt
[3] Faculty of Computers and Information, Cairo University, Giza, Egypt
[4] Scientific Research Group in Egypt (SRGE), Cairo, Egypt
http://www.egyptscience.net

Abstract. Fish image segmentation can be considered an essential process in developing a system for fish recognition. This task is challenging as different specimens, rotations, positions, illuminations, and backgrounds exist in fish images. In this research, a segmentation model is proposed for fish images using Salp Swarm Algorithm (SSA). The segmentation is formulated using Simple Linear Iterative Clustering (SLIC) method with initial parameters optimized by the SSA. The SLIC method is used to cluster image pixels to generate compact and nearly uniform superpixels. Finally, a thresholding using Otsu's method helped to produce satisfactory results of extracted fishes from the original images under different conditions. A fish dataset consisting of real-world images was tested. In experiments, the proposed model shows robustness for different cases compared to conventional work.

Keywords: Image segmentation · Fish database
Salp Swarm Algorithm · Superpixels · Optimization

1 Introduction

Meta-heuristics, in computer science and optimization, are procedures intended to generate, find or select heuristic partial search algorithms. That may provide a satisfactory solution to a problem with imperfect or incomplete information. Meta-heuristic can make a few assumptions about the optimization problem that needs to be solved. Therefore, it can be used for a diversity of problems compared to iterative methods and optimization algorithms. However, it does not assure that a globally optimal solution can be found on some problems. Many meta-heuristics apply a form of stochastic optimization. So, the solution found depends on a set of generated random variables in combinatorial optimization. By searching a large set of possible solutions, meta-heuristics can often find satisfactory solutions with less computational effort and optimization algorithms than simple heuristics or iterative methods.

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 42–51, 2018.
https://doi.org/10.1007/978-3-319-74690-6_5

Swarm algorithm is a category of meta-heuristic algorithms that simulate the collective behavior of decentralized self-organized natural or artificial systems. Many algorithms that belong to such category were found to be promising in solving challenging applications in a broad spectrum of fields. Those algorithms include Particle Swarm Optimization (PSO), Ant Colony Algorithm (ACO), Artificial Bee Colony (ABC), Grey Wolf Optimization (GWO) and Salp Swarm Algorithm (SSA). However, recent literature that compared a number of those algorithms demonstrated the superiority of SSA in single-objective optimization problems [1].

The research done by Ren et al. [2] proposed a new color image segmentation algorithm based on GrabCut. Their technique combined Bayes classification and Simple Linear Iterative Clustering (SLIC) followed by using the GrabCut method to obtain the segmentation. Clustering the features of a color image was done using the SLIC algorithm and the GrabCut framework to overcome the problem of the image segmentation deterioration problem. Another research by Wu et al. [3] presented an algorithm for applying cartoon image segmentation based on adaptive region propagation merging and SLIC superpixels. They proposed a method that improved the quality of the superpixels generation based on the connectivity constraint. Authors in [4] proposed a superpixel based model for thermal IR human face images. In [5], the segmentation algorithm based on SLIC superpixels had been used to eliminate the constructed defect and noise influence using the feature similarity in the preprocessing stage.

In [6], the authors proposed a new modeling method to model local session variations. They tested the efficiency of this approach using databases for fishes underwater that provide more session variations. In [7], the authors investigated the selected features subset for the binary classification problem using logistic regression model. They proposed a modified discrete PSO algorithm for feature selection problem. Their approach integrated an adaptive feature selection method that dynamically relied on the dependence and relevance of the features. The work in [8] proposed a new feature for fish age classification based on an ensemble of wrappers. The effectiveness of their approach using an Atlantic cod database was tested for various statistical learning classifiers.

The work presented in [9] proposed a new approach for feature selection based on the Fish School Search (FSS) optimization algorithm, which aimed to take into account premature convergence. The authors proposed binary encoding procedure for the internal mechanisms of fish school search. In [10] the authors combined K-means algorithm with mathematical morphology for fish images segmentation. Authors in [11] introduced a fish detection method using Bat optimization algorithm to reduce the time of classification within the fish detection process.

The authors in [12] developed a classifier for fish images recognition system that depended on color texture measurements extracted from gray level co-occurrence matrix. Their system started by acquiring an image containing fish pattern; then they segmented the image relying on color texture measurements. Authors in [13] proposed an automatic classification approach for the Nile Tilapia fish based on Support Vector Machines (SVMs) with feature extraction using Scale Invariant Feature Transform (SIFT) and Speeded Up Robust Features (SURF) algorithms. In [14] the centroid-contour distance method was used to classify fish species with two dorsal fins. Various image processing methods were applied on images to extract centroid contour distances. These distances were used as features, and the nearest neighbor algorithm was used for classification.

In this paper, a segmentation model was proposed for fish images using SSA. The segmentation was formulated using SLIC method whose parameters were optimized by SSA. The SLIC method clustered pixels for generating compact and nearly uniform superpixels. In experiments, the proposed model showed robustness for different cases.

2 Preliminaries

2.1 Salp Swarm Algorithm (SSA)

Salps tissues and movement are highly similar to jellyfishes. One of the most noticeable behaviors of salps is their swarming behavior. Salps form a swarm named salp chain. The main reason for this behavior for some researchers is done for achieving better locomotion using rapid, coordinated changes and foraging [1].

The work done by Mirjalili et al. [1] proposed a model of salp to solve optimization problems. They modeled the salp chains, by dividing the population into two categories: leader and followers. The leader is the salp at the front of the chain, whereas the rest of salps are considered as followers. The swarm is guided by the leader, while the followers follow each other. The positions of salps are defined in an n-dimensional search space where n is the number of variables of a given problem. Therefore, the positions of all salps are stored in a two-dimensional matrix called x. It is assumed that there is a food source called F in the search space as a target for the swarm. To update the leader's position, the following equation was applied

$$x_j^i = \begin{cases} F_j + c_l((ub_j - lb_j)c_2 + lb_j) \; c_3 \geqslant 0 \\ F_j - c_l((ub_j - lb_j)c_2 + lb_j) \; c_3 < 0 \end{cases} \tag{1}$$

where x_j^i shows the position of the leading salp in the j^{th} dimension, F_j represents the food source position in the j^{th} dimension, $(ub)_j$ indicates the upper bound of j^{th} dimension, $(lb)_j$ indicates the lower bound of j^{th} dimension, and the parameters c_2 and c_3 are randomly generated in the interval of $[0, 1]$. Equation (1) indicates that the leader updates its position according to the food source. The coefficient c_1 is a critical parameter in SSA because it balances exploration and exploitation defined as follows:

$$c_1 = 2e^{-(\frac{4l}{L})^2} \tag{2}$$

where l is the current iteration and L is the maximum number of iterations. The followers' positions are updated by the following equation:

$$x_j^i = \frac{1}{2}at^2 + v_0 t \tag{3}$$

where $i \geqslant 2$ and x_j^i shows the position of i^{th} follower salp in j^{th} dimension, t is time, v_0 is the initial speed, and $a = \frac{v_{final}}{v_0}$ where $v = \frac{x - x_0}{t}$. The discrepancy among iterations is equal to 1, and by considering $v_0 = 0$, this equation can be expressed as follows:

$$x_j^i = \frac{1}{2}(x_j^i + x_j^{i-1}) \tag{4}$$

where $i \geqslant 2$ and x_j^i shows the position of i^{th} follower salp in j^{th} dimension. With Eqs. (1) and (4), the salp chains can be simulated.

The SSA algorithm starts with initiating random positions for multiple salps. Then, the fitness of each salp is calculated to find the salp with the best fitness and assigned the position of the best salp to the variable F as the source food to be chased by the salp chain. In the meantime, the coefficient c_1 is updated using Eq. (2). For each dimension, the position of leading salp is updated using Eq. (1). Moreover, the position of follower salps is updated utilizing Eq. (4). If any of the salps goes outside the search space, it will be brought back on the boundaries. All the above steps except initialization need to be executed iteratively until the satisfaction of predefined criteria. Notice that, during optimization, the food source will be updated because the salp chain finds a better solution by exploiting and exploring the around space. The salp chain has the potential to move towards the global optimum that changes over the course of iterations. The SSA algorithm has the following features:

- SSA algorithm saves the best solution obtained so far and assigned it to the food source variable, so it never gets lost even if the whole population deteriorates.
- SSA algorithm updates the position of the leading salp concerning the food source which is the best solution obtained, so the leader always exploits and explores the around space.
- SSA algorithm updates the follower position of salps concerning each other. Thus they move gradually towards the leading salp.
- Gradual movements of follower slaps preserve the SSA algorithm from stagnating in the local optima.
- Parameter c_1 is the main controlling parameter, and it adaptively decreased during iterations. Thus, the SSA algorithm first explores the search space and then exploits it.

These features make the SSA algorithm theoretically and potentially able to solve single-objective optimization problems with unknown search spaces. The SSA algorithm computational complexity is of $O(t(d \times n + Cof \times n))$ where t represents the iterations number, d is the variables (dimension) number, n is the solutions number, and Cof indicates the objective function cost.

2.2 SLIC Method

Simple Linear Iterative Clustering (SLIC) is one of the most important superpixels segmentation algorithms that requires low computational power. To accurately generate compact and uniform superpixels, the algorithm combines 5-D colors and image plan space. SLIC algorithm performs local clustering in 5-D space which is the CIELAB colorspace of l, a, b values and the pixels coordinates of x, y [15]. Euclidean distances in CIELAB color space are useful for small distances. Measure of distance D_s is defined as

$$d_{lab} = \sqrt{((l_i - l_j)^2 + (a_i - a_j)^2 + (b_i - b_j)^2)}, \tag{5}$$

$$d_{xy} = \sqrt{((x_i - x_j)^2 + (y_i - y_j)^2)}, \tag{6}$$

$$D_s = d_{lab} + \frac{m}{S} d_{xy} \tag{7}$$

The algorithm starts with sampling cluster centers that are regularly spaced. Then, moves the centers to initialize locations according to the lowest gradient position in a 3 × 3 grid. Image gradients $G(x, y)$ are computed using

$$G(x, y) = \|I(x + 1, y) - I(x - 1, y)\|^2 + \|I(x, y + 1) - I(x, y - 1)\|^2 \tag{8}$$

where $I(x, y)$ is the lab vector of the pixel at position (x, y), and $\|.\|$ represents the $L2$ norm. Intensity information and color are taken into consideration. Finally, pixels in the larger segment neighborhood will have the same label. The algorithm then enforces relations by relabeling disjoint segments with the labels of the largest adjacent cluster.

3 Proposed Fish Segmentation Model

In this section, a model was proposed to extract fish from fish images under different conditions. This method uses the SLIC segmentation algorithm to produce superpixels based on the SSA optimization and then apply the method of Otsu to threshold the output superpixel image. Algorithm (1) presents the proposed fish segmentation method steps.

Algorithm 1. Fish Segmentation Proposed Model

```
Input the fish images $\{I\}_{i=1}^{N}$, where number of images is represented by $N$ and the $i^{th}$ input image is represented by $I_i$.
Calculate superpixels $S_i(I_i)$ using SLIC segmentation with initial values optimized by SSA.
Use the method of Otsu to separate the pixels in the foreground or background for superpixels image $S_i(I_i)$ of SLIC.
Convert superpixels' image to a binary image $B_i(S_i)$ using the optimum threshold.
Determine the fish pixels from the original image $I_i$ to get $I_i(fish)$ using the binary image $B_i(S_i)$.
```

4 Experimental Results and Discussion

In this paper, a dataset of 3,960 real-world fish images collected from 468 species was used [6]. These images were captured in different conditions, namely, “controlled”, “out-of-the-water” and “in-situ”. The “controlled” images were taken under a constant background and illumination is controlled consist. The “out-of-the-water” images were natural underwater images of fish. The “in-situ” ones were out of the water images, and the background is varying with limited control over the illumination. Figure 1 shows a sample for each condition.

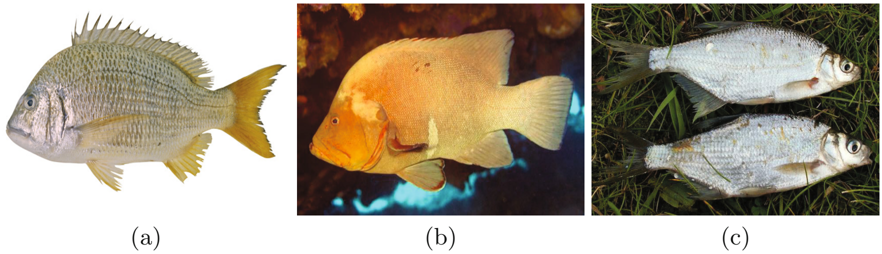
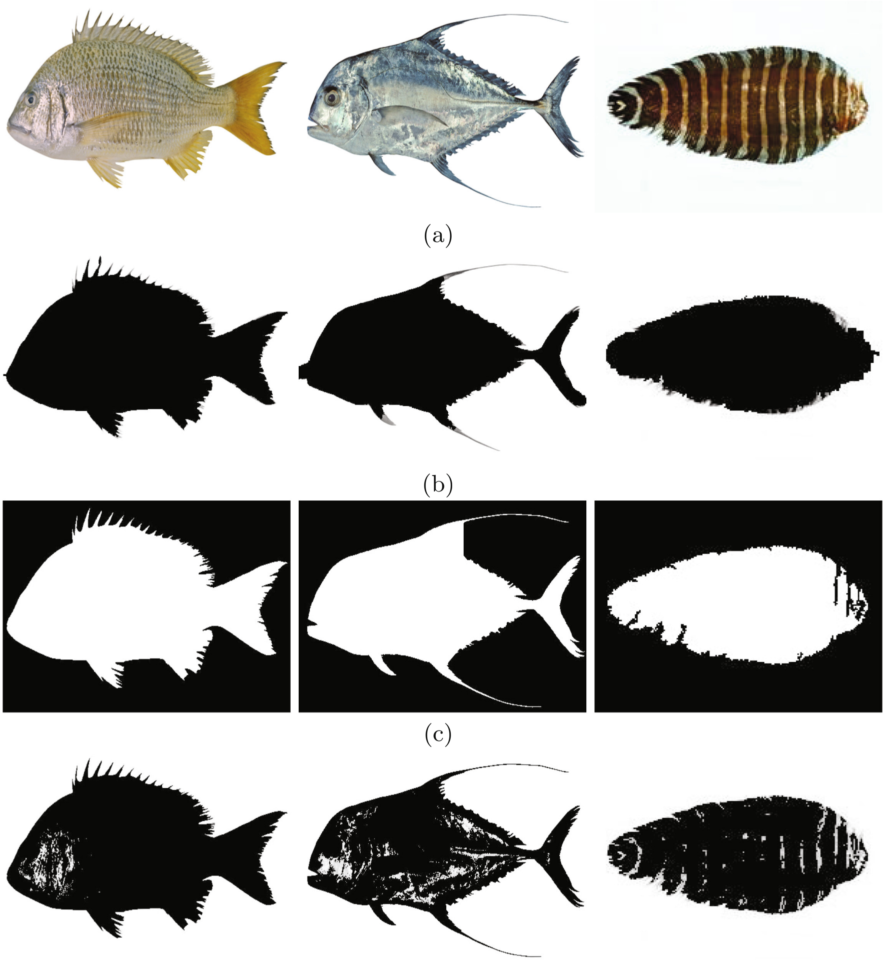

(a) (b) (c)

Fig. 1. Samples of fish images. (a) controlled (b) out-of-the-water (c) in-situ

The effectiveness of the proposed model was investigated by two types of analysis based on the applied conditions for the fish images in the dataset. In the first scenario, the proposed method was applied to the “controlled” images. Three different fish specimens with different colors, namely Acanthopagrus Latus, Alectis Ciliaris, and Aesopia Cornuta, captured under a constant background with controlled illumination were used in this scenario. Figure 2 shows the proposed segmentation results for different images. The results showed the robustness of this research method compared to a direct threshold using the Otsu’s method and the active counters [16,17] method.

The second scenario used fish images under “out-of-the-water” and “in-situ” conditions. These types of images were more complicated than the controlled ones since the background was not constant and the illumination might vary from one image to another. Figure 3 shows the results for Aethaloperca Rogaa and Acanthopagrus Australis fish specimens. In addition, a segmentation method based on active counters was implemented and its relevant results were compared with the proposed model. The same database was used for this comparison.



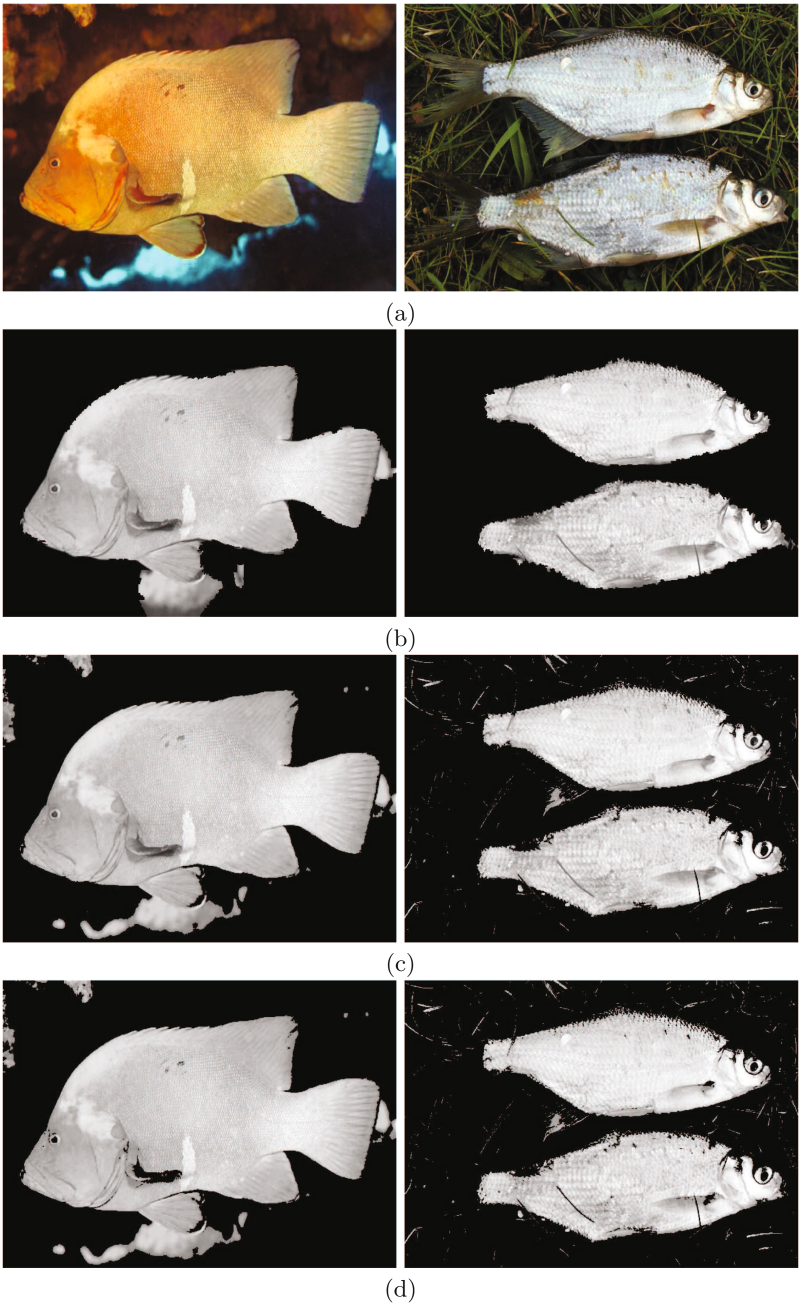
(a)

(b)

(c)

(d)

Fig. 2. Fish segmentation of the proposed method under "controlled" conditions. (a) Original images (b) Proposed segmentation (c) Active contours [16,17] (d) Otsu's segmentation



(a)

(b)

(c)

(d)

Fig. 3. Fish segmentation of the proposed method under "out-of-the-water" and "in-situ" conditions. (a) Original images (b) Proposed segmentation (c) Active contours [16, 17] (d) Otsu's segmentation

5 Conclusion

In this paper, a segmentation method for real-world fish images was proposed based on the SSA. The SLIC method was used with initial parameters optimized by the SSA to generate compact and uniform superpixels. A thresholding with a simple Otsu's method was then used for the superpixel image to segment fishes and excluded the background from the original images under different conditions. A fish dataset consisting of real-world images with more than 400 species was tested. Results showed that the proposed model showed a robustness for different cases compared to conventional work.

References

1. Mirjalili, S., Gandomi, A.H., Mirjalili, S.Z., Saremi, S., Faris, H., Mirjalili, S.M.: Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems. Adv. Eng. Softw. **114**, 163–191 (2017). https://doi.org/10.1016/j.advengsoft.2017.07.002
2. Ren, D., Jia, Z., Yang, J., Kasabov, N.K.: A practical grabcut color image segmentation based on bayes classification and simple linear iterative clustering. IEEE Access **5**, 18480–18487 (2017)
3. Wu, H., Wu, Y., Zhang, S., Li, P., Wen, Z.: Cartoon image segmentation based on improved SLIC superpixels and adaptive region propagation merging. In: Proceedings of IEEE International Conference on Signal and Image Processing (ICSIP), pp. 277–281. IEEE (2016)
4. Ibrahim, A., Gaber, T., Horiuchi, T., Snasel, V., Hassanien, A.E.: Human thermal face extraction based on superpixel technique. In: Proceedings of the 1st International Conference on Advanced Intelligent System and Informatics (AISI 2015), pp. 163–172. Springer International Publishing (2016)
5. Chen, X.-T., Zhang, F., Zhang, R.-Y.: Medical image segmentation based on SLIC superpixels model. In: Proceedings of SPIE International Conference on Innovative Optical Health Science, vol. 10245, pp. 10245–10248 (2017)
6. Anantharajah, K., Ge, Z., McCool, C., Denman, S., Fookes, C., Corke, P., Tjondronegoro, D., Sridharan, S.: Local inter-session variability modelling for object classification. In: Proceedings of IEEE Winter Conference on Applications of Computer Vision, pp. 309–316 (2014)
7. Unler, A., Murat, A.: A discrete particle swarm optimization method for feature selection in binary classification problems. Eur. J. Oper. Res. **206**(3), 528–539 (2010)
8. Bermejo, S.: Ensembles of wrappers for automated feature selection in fish age classification. Comput. Electron. Agric. **134**(Supplement C), 27–32 (2017)
9. Sargo, J.A.G., Vieira, S.M., Sousa, J.M.C., Filho, C.J.A.B.: Binary fish school search applied to feature selection: application to ICU readmissions. In: Proceedings of IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), pp. 1366–1373 (2014)
10. Yao, H., Duan, Q., Li, D., Wang, J.: An improved k-means clustering algorithm for fish image segmentation. Math. Comput. Model. **58**(3), 790–798 (2013)
11. Fouad, M.M., Zawbaa, H.M., Gaber, T., Snasel, V., Hassanien, A.E.: A fish detection approach based on bat algorithm. In: Proceedings of the 1st International Conference on Advanced Intelligent System and Informatics (AISI 2015), pp. 273–283. Springer International Publishing (2016)

12. Alsmadi, M., Omar, K., Noah, S., Almarashdeh, I.: Fish recognition based on robust features extraction from color texture measurements using back-propagation classifier. J. Theor. Appl. Inf. Technol. **18**(1), 11–18 (2010)
13. Fouad, M.M.M., Zawbaa, H.M., El-Bendary, N., Hassanien, A.E.: Automatic Nile Tilapia fish classification approach using machine learning techniques. In: 13th International Conference on Hybrid Intelligent Systems (HIS 2013), pp. 173–178 (2013)
14. Iscimen, B., Kutlu, Y., Uyan, A., Turan, C.: Classification of fish species with two dorsal fins using centroid-contour distance. In: 2015 23rd Signal Processing and Communications Applications Conference (SIU), pp. 1981–1984 (2015)
15. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Ssstrunk, S.: Slic superpixels compared to state-of-the-art superpixel methods. IEEE Trans. Pattern Anal. Mach. Intell. **34**(11), 2274–2282 (2012)
16. Chan, T.F., Vese, L.A.: Active contours without edges. IEEE Trans. Image Process. **10**(2), 266–277 (2001)
17. Filipe, S., Alexandre, L.A.: Algorithms for invariant long-wave infrared face segmentation: evaluation and comparison. Pattern Anal. Appl. **17**(4), 823–837 (2014)

Swarming Behaviors of Chicken for Predicting Posts on Facebook Branding Pages

Khaled Ahmed[1,3](✉), Aboul Ella Hassanien[1,3],
Ehab Ezzat[1], and Siddhartha Bhattacharyya[2,3]

[1] Faculty of Computers and Information, Cairo University, Giza, Egypt
khaled.elahmed@gmail.com, aboitcairo@gmail.com, e.ezat@fci-cu.edu.eg
[2] RCC Institute of Information Technology, Kolkata 700 015, India
dr.siddhartha.bhattacharyya@gmail.com
[3] Scientific Research Group in Egypt (SRGE), Cairo, Egypt
http://www.egyptscience.net

Abstract. The rapid increase in social networks data and users present an urgent need for predicting the performance of posted data over these networks. It helps in many industrial aspects such as election, public opinion detection and advertising or branding over social networks. This paper presents a new posts' prediction system for Facebook's branding pages concerning the user's attention and interaction. CSO is utilized to optimize the ANFIS parameters for accurate prediction. CSO-ANFIS is compared with several methods including ANFIS, particle swarm optimization, genetic algorithm and krill herd optimization.

Keywords: Facebook branding pages · Predicting posts metrics
Social network analysis · Chicken swarm optimization
Adaptive neuro-fuzzy inference system

1 Introduction

Social networks are connecting millions of users with their interactions and interests. Social networks analysis (SNA) is the process of analyzing and getting valuable insights from these networks [1]. SNA presents many indicators and advantages at social domain such as election [2], medical domain such as collecting patient experience [3] and in industrial domain such as collecting consumer experience and feedback on product or service by applying sentiment analysis on consumers' review [4], sharing products to cluster of users (branding pages' advertising) [5]. Targeting these users by an online post that contains an advertisement is an urgent, fast, a low cost and more efficient approach instead of the traditional TV advertising. Connecting companies' with their online users or customers is important to target [6]. Predicting Facebook's post performance metrics such as number of likes, number of shares, total reaches, impression (happy, sad, neutral, angry, ... etc.), comments, lifetime of post consumers, total interactions with the post and engaged users help decision makers or business owners to get more benefits, income, decide (when, how, where) the online post

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 52–61, 2018.
https://doi.org/10.1007/978-3-319-74690-6_6

will be suitable and more efficient. Some researchers tackle the problem of building a predictive model for social networks brand advertising's posts metrics effect based on traditional models such as statistical or voting techniques, which are no efficient in many cases [7]. On the other hand, another researchers tackled this problem by building an optimized model based on data mining techniques, ANFIS or hybrid swarm optimization algorithms with ANFIS for more accurate prediction results [8].

Basic ANFIS is used for prediction in literature [8], but its results need to be more accurate. In [9] presents particle swarm optimization with ANFIS (PSO-ANFIs), In [10] presents Genetic Algorithm with ANFIS GA-ANFIS, In [11] presents Krill-ANFIS for overcoming and optimizing the training parameters of ANFIS for better predication results. In this paper a new hybrid method is proposed; it combines the original ANFIS with CSO algorithm to predict posts' efficiency metrics for Facebook's branding pages. To investigate the effective of the proposed method, there are two experiments are applied using some benchmark data sets, then the Facebook metrics data set is used.

The rapid increasing on the internet and social network that led users to present big data of their status, hobbies, profiles, check-in places and interaction that make the business stack holder use online social networks pages for sharing their services and products to all users and wait for the responses or the interactions on these posts. Traditional prediction techniques for helping and advising business owner for his goods and services campaign are not efficient as hard copy, online voting systems, or statistical reports. There is an urgent need for a method with higher prediction results which near to real world to inform business owner information about the products posts on social networks, performance metrics, and how their online campaign will face and reach the attention of social network.

The contributions of this paper are CSO-ANFIS Overcomes the drawbacks of the original ANFIS and improves its accuracy for prediction as a new prediction method based on CSO and ANFIS. Our model is tested to determine the Facebook posts' efficiency to help decision maker and business owner for their posts on social networks. The remainder of this paper is arranged as follows, preliminaries are presented in Sect. 2. Section 2.1 states basic ANFIS. Section 2.2 explains chicken swarm algorithm. Section 3 introduces the proposed prediction system. Section 4 shows the experiments and discusses the results. The conclusion and future works are given in the last section.

2 Preliminaries

This section provides a brief explanation of the basic framework of Adaptive neuro-fuzzy inference system and chicken swarm optimization algorithm along with some of the key basic concepts.

2.1 Adaptive Neuro-Fuzzy Inference System

Adaptive neuro-fuzzy inference system (ANFIS) is a predictor model which is used widely in different domains such as industrial domain [12], wind speed [11], crop yield [13], stock market forecasting [14], social domain, flood forecasting [15], medical domain [16].

IF and *then* rules is the core of ANFIS [12], it uses this rule to map the input values and produce output values. There are five layers build ANFIS, the rule numbers 1 and 2 are explained in Eqs. (1) and (2) respectively [12].

Rule 1:

$$IFx\ isA_1 + y\ isB_1 \longrightarrow f1 = p1x + q1y + r1 \tag{1}$$

Rule 2:

$$IFx\ isA_2 + y\ isB_2 \longrightarrow f2 = p2x + q2y + r2 \tag{2}$$

where x and y has member function of $A_{1,2}$ and $B_{1,2}$ and the output of the functions are $p_{1,2}$ $q_{1,2}$,$r_{1,2}$.

In the layer (1): each node has its own output which is defined as:

$$O_{1i} = \mu_{A_i}(X), i = 1, 2, O_{2i} = \mu_{B_{1-2}}(Y), i = 3, 4 \tag{3}$$

Where x, y are the input to node i, and A_i, B_i are the members values of the member functions μ_A and μ_B, respectively. $\mu_{A_i}(X)$ and $\mu_{B_{i-2}}(Y)$ are the generalized Gaussian member function that declared by:

$$\mu(x) = e^{-(x-\frac{\rho_i}{\sigma_i})^2} \tag{4}$$

Where σ_i and ρ^i indicate the premise parameters set. Each node calculates the firing rule's strength in the next layer based on the following Equation.

$$O_{2i} = \mu_{Ai}(X) * \mu_{B_{1-2}}(Y) \tag{5}$$

The next layer calculates the normalized firing strength as:

$$O_{3i} = \varpi_i = \omega_i / \Sigma\omega_i \tag{6}$$

While the node in layer four is an adaptive node. Its output is calculated as:

$$O_{4,i} = \varpi_i f_i = \varpi_i(p_i x + q_i y + r_i) \tag{7}$$

Where p_i, q_i and r_i is the parameter set of the node. ϖ_i indicates the normalized firing strengths.

$$O_5 = \sum \varpi_i F_i \tag{8}$$

The final layer is a single node, and the output is determined as the summation of all incoming signals.

2.2 Chicken Swarm Optimization Algorithm

Chicken swarm optimization (CSO) is a nature inspired optimization algorithm which is used in many kinds of literature such as feature selection and enhancing parameter estimation [17,18]. CSO algorithm is consists of three basic steps check for the best fitness value which will be a rooster; the *worst* are chicks while the *rest* are hens. Each group is consist of two hen and one rooster with a set of chicks. Equations (9) and (10) are stating the movements of rooster while Eqs. (11), (13) and (14) are stating the behavior of Hen while Eq. (15) states the behavioral of chicks [19]. Rooster with best fitness values is defined by the following Equation.

$$X_{i,j}^{t+1} = X_{i,j}^{t} * (1 + Rand(0, \sigma^2)) \tag{9}$$

Where $X_{i,j}^{t}$ is old position of the rooster, $X_{i,j}^{t+1}$ is the new position, *Rand* is small value [zero, and $sigma^2$].

$$\sigma^2 = \begin{cases} {\scriptstyle 1,\ iff_i < f_k,} \\ \\ {\scriptstyle Exp}\dfrac{(f_k - f_i)}{abc((f_i) + \in)} \end{cases} \tag{10}$$

where rooster index is k, related position for rooster is i and $\in$ small integer.

The motion for hens is defined as the following:

$$X_{i,j}^{t+1} = X_{i,j}^{t} + S1 * Randd * (X_{r1,j}^{t} - X_{i,j}^{t}) + P \tag{11}$$

$$P = S2 * Randd * (X_{r2,j}^{t} - X_{i,j}^{t}) \tag{12}$$

$$S1 = Exp\frac{(f_k - f_{r1})}{abc((f_i) + \in)} \tag{13}$$

$$S2 = Exp(f_{r2} - f_i) \tag{14}$$

Where *Randd* is a random value over $[0, 1]$, r_1 is rooster index, r_2 hen index or rooster index which randomly chosen and $r_1 \sharp r_2$.

Chicks motion is defined as the following:

$$X_{i,j}^{t+1} = X_{i,j}^{t} + FL * (X_{m,j}^{t} - X_{i,j}^{t}) \tag{15}$$

Where FL is random value between [0.2] and m is the chick's mother index.

3 The Proposed Predicting Posts of Facebook Branding Pages

Chicken swarm optimization algorithm is used for optimizing the training parameters of basic ANFIS for classification problems and achieved promised results

[20]; our proposed method is a hybrid model that based on ANFIS and CSO algorithm, it is called CSO-ANFIS for prediction or regression problems. First divides the input data set into 70% for training the proposed method and the rest 30% for testing, then uses the training data set to learn and train CSO-ANFIS method, after that optimizes the parameters of the ANFIS using CSO algorithm by using the chicken positions in the population. The next step is to calculate fitness value for each swarm object of the chicken if the best fitness value updates its position using rooster Eq. 10, else if the fitness value is the worst update its position using hen Eq. 12 else update its position using chicks Eq. 16. Repeat until max iteration number is achieved and best solution is presented during these updates and finally measure the prediction using the test part of the data set and the performance metrics which is presented in Fig. 1.

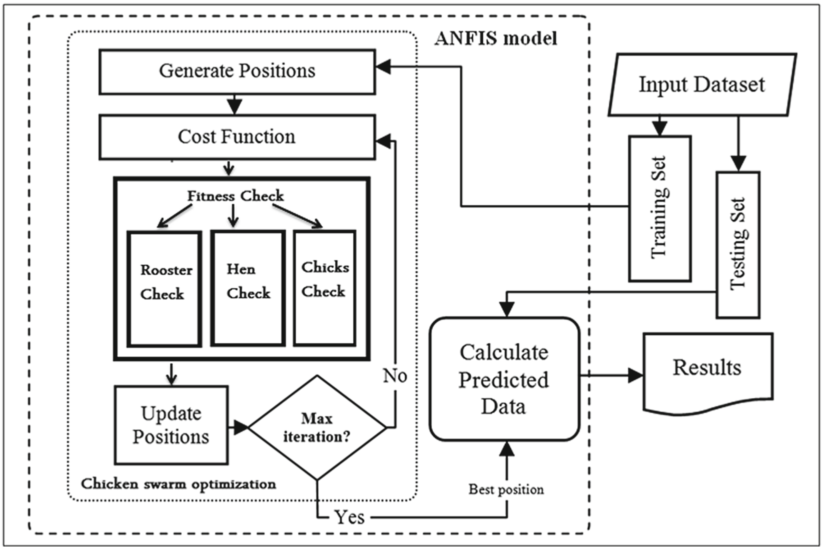


Fig. 1. CSO-ANFIS structure.

CSO-ANFIS is a highly accurate prediction model and an important industrial approach as well, which helps manager, business owner or stake holder to achieve more social media users' attention to online products' posts. Figure 2 presents the steps and hierarchy of how to use CSO-ANFIS, which starts with manager or business owner types a post on his online brand page, our method generates for this post an input vector (posted, permanent link, post ID, Post message, type, category, and paid) to be entered to our proposed method to predict the performance metric for this post.

Predicted performance metric is 12 items individually can be predicted and generated from CSO-ANFIS such as (Lifetime post total reach, Lifetime post total impressions, Lifetime engaged users, Lifetime post consumers, Lifetime post consumption, lifetime people who have liked a page and engaged with a post, lifetime post reach by people who like a page, Lifetime post impressions, likes, comments, shares and Total interactions). The manager checks the results of metrics and decides: *if* the predicted results are high *then* publish the post *else* modify the post before real submitting to social network and re-post again to restart this cycle; finally, our method helps in publishing posts with high ranked prediction performance metrics.

This workflow helps manager to answer three questions such as: when does the post efficient? Is this content efficient? Where to share the post? If the predicted metrics results are low, the manager decides to wait or modify the content or choose the other region to share the post to and recheck by CSO-ANFIS. CSO-ANFIS can predict the interactions metrics and can help as well to deliver how online interactions will be with social networks' advertisements. CSO-ANFIS helps as well manager to share the correct post if this post has high-performance metrics. Figure 2 presents the workflow of the model 2.

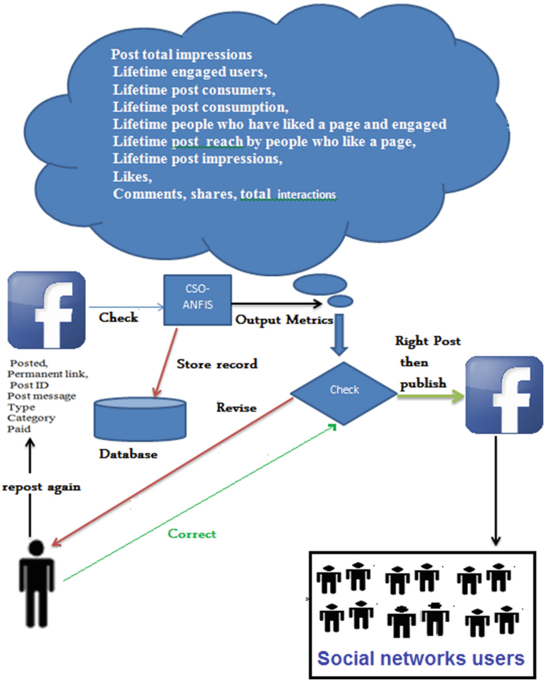


Fig. 2. CSO-ANFIS steps for business owner or manager.

4 Experiments and Discussion

The experiments are executed over Facebook's renowned cosmetic page posts efficiency metrics which is round 790 posts that have published at 2014. This data set consists of seven input features for the post (posted, permanent link, post message, post ID, type, category, paid) and twelve features such as (Lifetime post total reach, Lifetime engaged users, Lifetime post total impressions, Lifetime post consumption, Lifetime post consumers, lifetime people who have liked a page and engaged with a post, lifetime post reach by people who like a page, Lifetime post impressions, likes, comments, shares and Total interactions) for measuring the impact on social network used as prediction features [21]. The experiments used Lifetime Post Consumers as the predicted feature for evaluating the accuracy of the proposed method. Experiments are executed using random 70% of each data set, while the rest of the data sets 30% for testing. Initialization parameters are population size of 25, the lower bound is −10, the upper bound is 10, and the maximum number of iterations is 100, The experiments are executed using "Matlab 2014" on "Windows 10". The parameters setting of all algorithms is shown in Table 1.

Table 1. The parameters setting of all algorithms

The algorithm	The parameters setting
CSO	G = 10, rPercent = 0.15, hPercent = 0.7, and mPercent = 0.25
krill	Vf = 0.02, Dmax = 0.09, and Nmax = 0.1
PSO	Inertia Weight Damping Ratio = 0.99, Inertia Weight = 1, Global Learning Coefficient = 2, and Personal Learning Coefficient = 1
GA	Crossover Percentage = 0.4, gamma = 0.7, Mutation Percentage = 0.7, Mutation Rate = 0.15, and Selection Pressure(beta) = 8
ANFIS	ErrorGoal = 0; InitialStepSize = 0.01

The CSO-ANFIS performance is evaluated by Root Mean Square Error (RMSE) and Average Absolute Percent Relative Error (AAPRE) as in Eqs. 16 and 17 [11].

$$RMSE = \sqrt{\frac{1}{n}\Sigma_1^n (y_i - Y_i)^2} \tag{16}$$

$$AAPRE = \frac{100}{N}(\sum_{i=1}^{N} |\frac{(x_i - y_i)}{y_i}|) \tag{17}$$

The experimental results proved that CSO-ANIS achieved higher prediction results than original ANFIS. That due to using of CSO algorithm for training ANFIS parameters which led to better accuracy of the prediction. The results of CSO-ANFIS were compared against four models, namely, original ANFIS, PSO-ANFIS [9], GA-ANFIS [10], and Krill-ANFIS [11]. CSO-ANFIS proved the good results in term of RMSE and AAPRE. Experimental results of RMSE and AAPRE are illustrated in Fig. 3 and are presented in Table 2.

Table 2. The average of quality measures over 20 runs for Facebook metrics data set.

Model	RMSE	AAPRE
CSO-ANFIS	0.07531	1.6154
ANFIS	0.08521	1.6147
Krill-ANFIS	0.0791	1.6294
PSO-ANFIS	0.07686	1.6262
GA-ANFIS	0.07644	1.5843

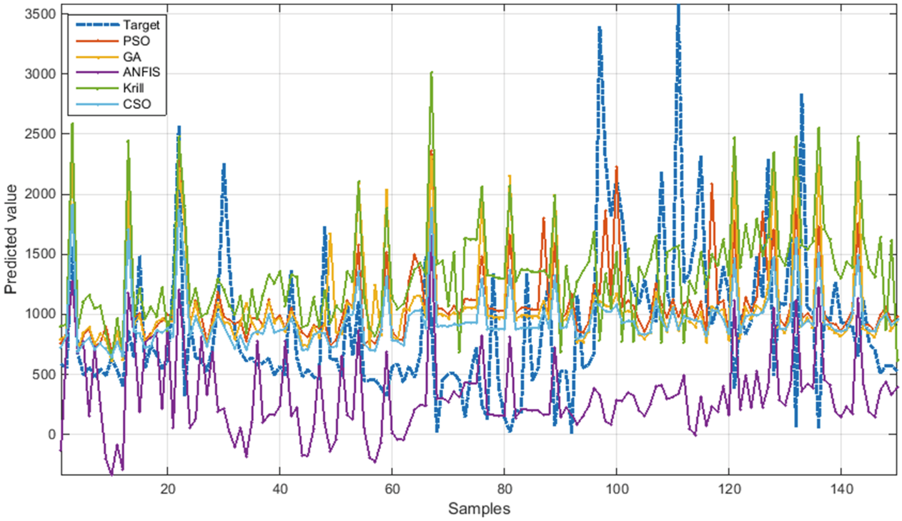


Fig. 3. Experimental results over all measures.

5 Conclusion and Future Work

This paper presents a new method CSO-ANFIS. It uses as a prediction method. It achieved good results than four well-known algorithms, namely, original ANFIS, particle swarm optimization, Genetic Algorithm and Krill Herd concerning quality measures of RMSE and AAPRE. Facebook metrics data set and yielded good results than all algorithms. CSO-ANFIS presented a prediction method that can be used effectively with Facebook metrics of brand posts on the social network which helps business stake holders to decide (when, where, and content of advertisements). Future works are to increase our method accuracy by integrating sentiment analysis layer and applying chaotic swarm with ANFIS.

References

1. Kazienko, P., Chawla, N.: Applications of Social Media and Social Network Analysis. Springer, Cham (2015)
2. Wang, H., Can, D., Kazemzadeh, A., Bar, F., Narayanan, S.: A system for real-time twitter sentiment analysis of 2012 U.S. presidential election cycle. In: Proceedings of the ACL 2012 System Demonstrations, ACL 2012, Stroudsburg, PA, USA, pp. 115–120. Association for Computational Linguistics (2012)
3. Greaves, F., Ramirez-Cano, D., Millett, C., Darzi, A., Donaldson, L.: Harnessing the cloud of patient experience: using social media to detect poor quality healthcare. BMJ Qual. Saf. **22**(3), 251–255 (2013)
4. Fang, X., Zhan, J.: Sentiment analysis using product review data. J. Big Data **2**(1), 5 (2015)
5. Kohli, C., Suri, R., Kapoor, A.: Will social media kill branding? Bus. Horiz. **58**(1), 35–44 (2015)
6. Stavrianea, A., Kavoura, A., Giannakopoulos, G., Sakas, D.P., Kyriaki-Manessi, D.: Social media's and online user-generated content's role in services advertising. In: AIP Conference Proceedings, vol. 1644, pp. 318–324. AIP (2015)
7. Fan, W., Gordon, M.D.: The power of social media analytics. Commun. ACM **57**(6), 74–81 (2014)
8. Bastani, S., Jafarabad, A.K., Zarandi, M.H.F.: Fuzzy models for link prediction in social networks. Int. J. Intell. Syst. **28**(8), 768–786 (2013)
9. Rini, D.P., Shamsuddin, S.M., Yuhaniz, S.S.: Particle swarm optimization for ANFIS interpretability and accuracy. Soft Comput. **20**(1), 251–262 (2016)
10. Morteza Zanaganeh, S., Mousavi, J., Shahidi, A.F.E.: A hybrid genetic algorithmadaptive network-based fuzzy inference system in prediction of wave parameters. Eng. Appl. Artif. Intell. **22**(8), 1194–1202 (2009)
11. Ahmed, K., Ewees, A.A., El Aziz, M.A., Hassanien, A.E., Gaber, T., Tsai, P.-W., Pan, J.-S.: A Hybrid Krill-ANFIS Model for Wind Speed Forecasting, pp. 365–372. Springer, Cham (2017)
12. Jang, J.-S.R.: ANFIS: adaptive-network-based fuzzy inference system. IEEE Trans. Syst. Man Cybern. **23**(3), 665–685 (1993)
13. Naderloo, L., Alimardani, R., Omid, M., Sarmadian, F., Javadikia, P., Torabi, M.Y., Alimardani, F.: Application of ANFIS to predict crop yield based on different energy inputs. Measurement **45**(6), 1406–1413 (2012)
14. Boyacioglu, M.A., Avci, D.: An adaptive network-based fuzzy inference system (ANFIS) for the prediction of stock market return: the case of the Istanbul stock exchange. Expert Syst. Appl. **37**(12), 7908–7912 (2010)
15. Rezaeianzadeh, M., Tabari, H., Yazdi, A.A., Isik, S., Kalin, L., Kalin, L.: Flood flow forecasting using ANN, ANFIS and regression models. Neural Comput. Appl. **25**(1), 25–37 (2014)
16. Ziasabounchi, N., Askerzade, I.: ANFIS based classification model for heart disease prediction. Int. J. Eng. Comput. Sci. **14**, 7–12 (2014)
17. Hafez, A.I., Zawbaa, H.M., Emary, E., Mahmoud, H.A., Hassanien, A.E.: An innovative approach for feature selection based on chicken swarm optimization. In: 2015 7th International Conference of Soft Computing and Pattern Recognition (SoCPaR), pp. 19–24. IEEE (2015)
18. Chen, S., Yan, R.: Parameter estimation for chaotic systems based on improved boundary chicken swarm optimization. In: International Symposium on Optoelectronic Technology and Application 2016, p. 101571K. International Society for Optics and Photonics (2016)

19. Meng, X., Liu, Y., Gao, X., Zhang, H.: A new bio-inspired algorithm: chicken swarm optimization. In: International Conference in Swarm Intelligence, pp. 86–94. Springer (2014)
20. Roslina, Zarlis, M., Yanto, I.T.R., Hartama, D.: A framework of training ANFIS using chicken swarm optimization for solving classification problems. In: 2016 International Conference on Informatics and Computing (ICIC), pp. 437–441, October 2016
21. Lichman, M.: UCI Machine Learning Repository (2013). http://archive.ics.uci.edu/ml

Enhancing AGDE Algorithm Using Population Size Reduction for Global Numerical Optimization

Ali Khater Mohamed[1](✉), Ali Wagdy Mohamed[2](✉), Ehab Zaki Elfeky[1], and Mohamed Saleh[1]

[1] Decision Support Department, Faculty of Computers and Information,
Cairo University, Giza 12613, Egypt
ak.mohamed@mu.edu.sa,
{e.elfeky,m.saleh}@fci-cu.edu.eg

[2] Operations Research Department, Institute of Statistical Studies and Research,
Cairo University, Giza 12613, Egypt
aliwagdy@gmail.com

Abstract. Adaptive guided differential evolution algorithm (AGDE) is a DE algorithm that utilizes the information of good and bad vectors in the population, it introduced a novel mutation rule in order to balance effectively the exploration and exploitation tradeoffs. It divided the population into three clusters (best, better and worst) with sizes 100p%, NP-2 * 100p% and 100p% respectively. Where p is the proportion of the partition with respect to the total number of individuals in the population (NP). AGDE selects three random individuals, one of each partition to implement the mutation process. Besides, a novel adaptation scheme was proposed in order to update the value of crossover rate without previous knowledge about the characteristics of the problems. This paper introduces enhanced AGDE (EAGDE) with non-linear population size reduction, which gradually decreases the population size according to a non-linear function. Moreover, a newly developed rule developed to determine the initial population size, that is related to the dimensionality of the problems.

The performance of the proposed algorithm is evaluated using CEC2013 benchmarks and the results are compared with the state-of-art DE and non-DE algorithms, the results showed a great competitiveness for the proposed algorithm over the other algorithms, and the original AGDE.

Keywords: Differential Evolution · Novel mutation · Adaptive crossover
Initial population · Population reduction

1 Introduction

Differential Evolution (DE) is a population-based heuristic algorithm proposed by Storn and Price [1] to solve global optimization problems with different characteristics over continuous space. Despite its simplicity, it proved a great performance in solving non-differentiable, non-continuous and multi-modal optimization problems [2]. DE has three main control parameters which are the crossover (*CR*), mutation factor (*F*) and

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 62–72, 2018.
https://doi.org/10.1007/978-3-319-74690-6_7

population size (*NP*). The values of the control parameters affect significantly on the performance of DE. Therefore, the tuning of those control parameters is considered a challenging task. DE has a great performance in exploring the solution space and this is considered as the main advantage, on the other side, an obvious weak point is its poor performance in exploitation phase which may cause a stagnation and/or premature convergence. During the last two decades, the problem of finding the balance between the exploration and exploitation has attracted many researchers in order to improve the performance of DE by developing new mutation strategies or hybridizing promising mutation strategies.

Das et al. [3] proposed an improved variant of DE/target-to-best/1/bin based on the concept of population members' neighborhood. Zhang and Sanderson [4] proposed a new mutation strategy "DE/current-to-pbest" with an optional external archive that utilizes the historical data in order to progress towards the promising direction and called it JADE. Qin et al. [5] proposed SaDE, in which a self-adaptive mechanism for trial vector generation is presented, that is based on the idea of learning from the past experience in generating promising solutions. Mohamed et al. [6–8] proposed a novel mutation strategy which is based on the weighted difference between the best and the worst individual during a specific generation, the new mutation strategy is combined with the basic mutation DE/rand/1/bin with equal probability for selecting each of them. Li and Yin [9] used two mutation strategies based on the best and random vectors. Mohamed [10] proposed IDE, in which new triangular mutation rule that selects three random vectors and adding the difference vector between the best and worst to the better vector. The new mutation rule is combined with the basic mutation rule through a non-linear decreasing probability rule. And a restart mechanism to avoid the premature convergence is presented. Recently, triangular mutation has been also used to solve IEEE CEC 2013 unconstrained problems [11], constrained non-linear integer and mixed-integer global optimization problems [12], IEEE CEC 2006 constrained optimization problems [13], CEC 2010 large-scale optimization problems [14], and stochastic programming problems [15].

Extensive research was presented for controlling the parameters, as control parameters play a vital role in the evolution process. Brest et al. [16] presented a new self-adaptive technique for controlling the parameters. Noman and Iba [17] proposed an adaptive crossover based on local search and the length of the search was adjusted using hill-climbing. Peng et al. [18] proposed rJADE, in which a weighting strategy is added to JADE, with a "restart with knowledge transfer" method in order to benefit from the knowledge obtained from the previous failure. Montgomery and Chen [19] presented a complete analysis of how much the evolution process affected by the value of CR. Mallipeddi et al. [20] proposed a pool of values for each control parameter to select the appropriate value during the evolution process. Wang et al. [21] proposed a new method that randomly chooses from a pool that contains three strategies in order to generate the trial vector and three control parameter settings, they called it CoDE. Yong et al. [22] presented CoBiDE, in which a covariance matrix learning for the crossover operator and a bimodal distribution parameter to control the parameters are introduced. Draa et al. [23] introduced a new sinusoidal formula in order to adjust the values of crossover and the scaling factor, they called it SinDE. A complete review could be found in [24, 25].

DE mechanism depends on selecting three random individuals from the population to perform the mutation process. Therefore, the population size must be greater than the selected vectors. Large population size increases the diversity but consumes more resources (function calls), while small population size may cause stagnation or tripping in local optima. Thus, the choosing of the population size is considered a very critical aspect. From the literature, it has been found that researchers choose the population size in four different ways. (1) choosing the population size for each problem separately based on the experience or previous knowledge and keep it constant during all runs [26, 27]. (2) relate the population size to the problem dimensionality [6, 28, 29]. (3) setting the population size fixed during all runs and independent of the dimension of the problems [22, 30]. (4) allowing the population size to vary during the runs using adaptation rule [31, 32]. A complete review of population size could be found in [33]. Based on the above review, this paper presents a non-linear population reduction technique and an initial population rule in order to enhance the AGDE algorithm presented by Mohamed and Mohamed [34]. The proposed non-linear reduction technique is a special case of Laredo et al. [35] which reduces the population size linearly. The proposed EAGDE is evaluated using CEC’2013 benchmark functions [36], and the results are compared with the AGDE and five state-of-art optimization algorithms: “The super-fit multi criteria adaptive differential evolution (SMADE), covariance matrix adaptive evolution strategy (CMAES), the self-adaptive Differential Evolution with PBX crossover (MDE_PBX), the Cooperatively Coevolving Particle Swarms Optimizers (CCPSO2) and Creatively-oriented optimization algorithm (COOA)” [34]. The next section introduces briefly the AGDE and the enhanced AGDE, the initial population size rule and the non-linear reduction formula. The experimental analysis is introduced in Sect. 3. And finally, the paper is concluded in Sect. 4 and future research points are outlined.

2 Adaptive Guided Differential Evolution with Non-linear Population Reduction

This section presents the AGDE in brief, the initial population size selection rule and non-linear reduction formula to enhance the performance of AGDE.

2.1 Novel Mutation Strategy

Adaptive guided differential evolution algorithm (AGDE) is a DE algorithm that utilizes the information of good and bad vectors in the population, it introduced a novel mutation rule in order to balance effectively the exploration and exploitation tradeoffs. It divided the population into three clusters (best, better and worst) with sizes 100p%, NP-2*100p% and 100p% respectively. Where p is the proportion of the partition with respect to the total number of individuals in the population (NP). AGDE selects three random individuals, one of each partition to implement the mutation process according to Eq. 1.

$$v_i^{G+1} = x_r^G + F \cdot (x_{p\_best}^G - x_{p\_worst}^G) \qquad (1)$$

2.2 New Adaptive Crossover

A novel adaptation scheme was proposed by Mohamed and Mohamed [34] in order to update the value of crossover rate without previous knowledge about the characteristics of the problems. A pool of promising crossover intervals is generated and an updated probability for each interval is calculated each generation based on the success of each interval (the offspring is fittest than its parent). The new adaptive crossover proved a good performance with an improved solution quality.

2.3 Setting the Initial Population Size

As mentioned before, the effect of choosing the population size affects highly on the performance of DE. Large population size increases the diversity but consumes more resources (function calls), while small population size may lead to local optima or premature convergence. Therefore, the experiment was conducted with initial NP = 5D, 10D, 15D, 20D and 25D to solve 10D, 30D and 50D problems and the best results were recorded to determine the best value of initial NP for each dimension. The best initial NP for 10D, 30D, and 50D problems are 10D, 6D and 4D respectively.

2.4 AGDE with Non-linear Population Size Reduction

As aforementioned, the population size reduction increases the performance of DE during the early stage of the optimization process. Large population size is needed for the exploration phase, then decreasing the population size is required in order to refine the solution quality. Laredo et al. [35] proposed a general framework for reducing the population size, so we modify the framework to reduce the population size in a non-linear manner. Equation 2 presents the proposed framework.

$$NP_{G+1} = round\left[NP_{initial} + (N_{min} - NP_{initial}) * \left(\frac{currFE}{maxFE}\right)^{X}\right] \qquad (2)$$

Where NP_{G+1} is the new population size in the next generation, *currFE* is the current function evaluation (FE), *maxFE* is the maximum allowed FE, $NP_{initial}$ is the initial population size as mentioned in Sect. 2.4, N_{min} is the minimum population size and is set to 12 in this study in order to obtain **at least** 2 elements in the best and worst partitions and X is the power that controls the linearity of the equation. The main idea behind designing population reduction technique is based on get rid of or delete greatly the worst solutions without affecting the exploration capability of the algorithm during early-middle stages of the optimization process while delete the worst solutions slightly during middle-later stages of the optimization process to enhance the exploitation tendency of the algorithm. i.e. the rate of deleting worst individuals in the first 50% of FEs is faster than the rate in the second 50% of FEs. Therefore, it can be clearly seen from Fig. 1 that the rate of deleting the worst individuals using the power $X = 1 - \left(\frac{currFE}{maxFE}\right)$ has a medium step length because it is faster than powers 1 and 2 and slower than 0.5, in the first 50% of FEs to keep the exploration capability of the algorithm. However, it is slower than all powers, with larger step length, in the second 50% of FEs to enhance the exploitation tendency of the algorithm by benefits from promising solutions as long as possible. Therefore, the power of $X = 1 - \left(\frac{currFE}{maxFE}\right)$ satisfies this proposed idea more than other curves with X = 0.5, 1 and 2 as indicated in Fig. 1. Therefore, an extensive experiment was conducted to obtain the best value for X by setting X = 0.5, 1, 2 and $1 - \left(\frac{currFE}{maxFE}\right)$, by solving the CEC’2013 benchmark problems for 10D, 30D and 50D. The results include the mean and standard deviation for each value of X are listed in Tables 1, 2 and 3, the best results are in bold.

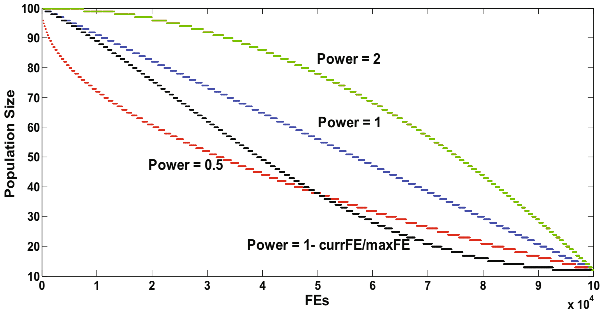


Fig. 1. Different values of X

Table 1. Results of 10D benchmark functions with different values of X

problem	AGDE	X = 1	X = [1-currFE/maxFE]	X = 0.5	X = 2
1	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
2	4.88E-06±3.34E-05	**0.00E+00±0.00E+00**	8.17E-07±2.12E-06	1.12E-06±5.96E-05	**0.00E+00±0.00E+00**
3	1.60E-02±4.15E-02	**1.90E-01±8.90E-01**	2.35E-01±5.86E-01	4.83E-01±1.50E+00	2.07E-01±9.05E-01
4	3.50E-03±1.68E-02	3.50E-03±1.68E-02	**0.00E+00±0.00E+00**	1.72E-07±5.89E-07	**0.00E+00±0.00E+00**
5	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
6	3.66E+00±4.79E+00	2.12E+00±4.08E+00	1.73E+00±3.78E+00	2.12E+00±4.08E+00	**5.77E-01±2.33E+00**
7	8.03E-03±2.06E-02	**2.64E-03±3.03E-03**	8.40E-03±1.57E-02	4.08E-03±9.38E-03	4.99E-03±6.01E-03
8	**2.03E+01±6.80E-02**	2.04E+01±7.51E-02	2.04E+01±6.92E-02	**2.03E+01±7.20E-02**	**2.03E+01±8.35E-02**
9	1.96E+00±1.64E+00	**1.32E+00±1.52E+00**	1.38E+00±1.20E+00	1.53E+00±1.47E+00	2.23E+00±1.51E+00
10	4.41E-02±2.32E-02	**2.77E-02±1.67E-02**	3.57E-02±2.72E-02	3.59E-02±1.86E-02	2.96E-02±1.80E-02
11	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
12	9.46E+00±2.23E+00	7.40E+00±2.43E+00	**6.54E+00±2.00E+00**	6.61E+00±1.90E+00	8.85E+00±2.27E+00
13	1.12E+01±4.18E+00	9.30E+00±3.62E+00	8.45E+00±3.84E+00	**8.40E+00±3.98E+00**	9.01E+00±3.14E+00
14	1.22E-02±2.50E-02	**1.22E-03±8.75E-03**	3.67E-03±1.48E-02	6.12E-03±1.88E-02	**1.22E-03±8.75E-03**
15	8.59E+02±1.66E+02	7.83E+02±1.44E+02	7.27E+02±1.56E+02	**7.01E+02±1.64E+02**	7.77E+02±1.37E+02
16	1.06E+00±1.72E-01	1.01E+00±1.84E-01	**9.59E-01±1.88E-01**	1.01E+00±1.91E-01	1.02E+00±1.77E-01
17	1.03E+01±7.17E-01	**1.01E+01±0.00E+00**	**1.01E+01±2.54E-04**	**1.01E+01±2.76E-03**	**1.01E+01±2.54E-04**
18	3.00E+01±3.21E+00	2.78E+01±3.46E+00	**2.63E+01±3.71E+00**	2.66E+01±2.96E+00	2.88E+01±3.20E+00
19	3.90E-01±6.40E-02	3.61E-01±4.94E-02	**3.18E-01±6.50E-02**	3.44E-01±6.16E-02	3.80E-01±7.57E-02
20	2.40E+00±4.06E-01	2.25E+00±2.71E-01	2.14E+00±2.74E-01	**2.08E+00±3.69E-01**	2.32E+00±2.29E-01
21	3.77E+02±7.64E+01	4.00E+02±0.00E+00	3.81E+02±6.01E+01	3.86E+02±5.67E+01	**3.73E+02±6.96E+01**
22	4.57E+01±1.64E+01	7.39E+00±3.39E+00	1.16E+01±1.47E+01	9.29E+00±3.67E+00	**8.13E+00±5.74E+00**
23	8.27E+02±1.62E+02	7.20E+02±1.53E+02	**6.39E+02±1.78E+02**	6.68E+02±1.92E+02	7.87E+02±1.47E+02
24	2.00E+02±1.13E+01	2.00E+02±9.39E-01	**1.97E+02±1.72E+01**	2.00E+02±1.68E+00	2.00E+02±1.15E+00
25	2.00E+02±2.77E-02	1.98E+02±1.22E+01	**1.96E+02±1.81E+01**	1.98E+02±1.28E+01	2.00E+02±1.52E-02
26	1.32E+02±3.31E+01	1.24E+02±3.32E+01	**1.21E+02±3.20E+01**	1.24E+02±3.40E+01	1.24E+02±3.11E+01
27	**3.00E+02±2.05E-03**	3.04E+02±1.96E+01	**3.00E+02±2.16E-03**	3.02E+02±1.40E+01	**3.00E+02±2.54E-03**
28	2.96E+02±2.80E+01	2.92E+02±3.92E+01	2.92E+02±3.92E+01	**2.88E+02±4.75E+01**	2.92E+02±3.92E+01

Table 2. Results of 30D benchmark functions with different values of X

problem	AGDE	X = 1	X = [1-currFE/maxFE]	X = 0.5	X = 2
1	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
2	**2.64E+04±1.43E+04**	5.49E+04±2.86E+04	5.04E+04±3.13E+04	5.06E+04±2.90E+04	7.39E+04±5.12E+04
3	3.07E+05±1.05E+06	**6.26E+00±2.56E+01**	2.15E+03±9.62E+03	5.07E+02±2.62E+03	6.41E+00±1.77E+01
4	**2.32E+00±3.59E+00**	4.11E+00±4.91E+00	3.85E+00±4.11E+00	3.32E+00±4.34E+00	4.52E+00±4.22E+00
5	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
6	1.55E+00±6.28E+00	2.49E+00±6.05E+00	4.84E+00±3.25E+00	**1.03E+00±7.89E-01**	2.13E+00±9.73E-01
7	3.98E+00±4.22E+00	**1.25E+00±9.56E-01**	2.34E+00±1.81E+00	1.32E+00±1.29E+00	2.05E+00±1.02E+00
8	**2.09E+01±4.67E-02**	2.10E+01±4.44E-02	**2.09E+01±4.94E-02**	**2.09E+01±4.09E-02**	**2.09E+01±5.17E-02**
9	**2.54E+01±5.64E+00**	2.78E+01±2.57E+00	2.56E+01±3.12E+00	2.63E+01±3.17E+00	2.89E+01±1.52E+00
10	3.11E-02±2.27E-02	1.09E-02±8.72E-03	1.34E-02±8.76E-03	1.33E-02±8.69E-03	**7.83E-03±6.82E-03**
11	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
12	9.45E+01±2.29E+01	1.05E+02±1.26E+01	**8.64E+01±1.50E+01**	**8.64E+01±2.34E+01**	1.22E+02±1.24E+01
13	1.10E+02±2.19E+01	1.28E+02±1.44E+01	**1.04E+02±1.54E+01**	1.13E+02±1.60E+01	1.37E+02±1.24E+01
14	**5.47E-02±3.25E-02**	1.23E-01±5.13E-02	2.00E-01±2.62E-01	1.06E-01±1.58E-01	9.10E+00±5.20E+00
15	5.99E+03±4.03E+02	6.08E+03±3.27E+02	**5.64E+03±3.69E+02**	5.87E+03±3.20E+02	6.17E+03±3.41E+02
16	**2.14E+00±2.61E-01**	2.34E+00±2.87E-01	2.27E+00±2.96E-01	2.35E+00±2.89E-01	2.39E+00±2.21E-01
17	**3.04E+01±2.12E-03**	**3.04E+01±7.32E-03**	**3.04E+01±1.71E-02**	**3.04E+01±8.40E-03**	3.12E+01±3.63E-01
18	1.84E+02±1.40E+01	1.95E+02±9.25E+00	**1.80E+02±1.31E+01**	1.86E+02±1.21E+01	2.00E+02±1.21E+01
19	3.07E+00±2.60E-01	3.16E+00±2.51E-01	**2.72E+00±1.97E-01**	2.87E+00±2.51E-01	3.50E+00±2.88E-01
20	**1.15E+01±3.72E-01**	1.18E+01±2.63E-01	1.16E+01±3.09E-01	1.17E+01±3.46E-01	1.19E+01±2.58E-01
21	3.00E+02±7.94E+01	**2.91E+02±6.61E+01**	2.94E+02±5.84E+01	3.06E+02±6.52E+01	2.95E+02±6.82E+01
22	6.12E+02±1.07E+02	1.12E+02±2.63E+00	**1.11E+02±2.90E+00**	**1.11E+02±2.82E+00**	1.18E+02±7.89E+00
23	6.14E+03±4.58E+02	6.10E+03±4.13E+02	**5.75E+03±4.48E+02**	5.95E+03±3.58E+02	6.41E+03±3.43E+02
24	2.08E+02±8.57E+00	**2.03E+02±3.98E+00**	2.05E+02±5.57E+00	2.04E+02±4.72E+00	**2.03E+02±2.70E+00**
25	**2.76E+02±1.43E+01**	2.85E+02±7.39E+00	2.80E+02±8.28E+00	2.83E+02±7.27E+00	2.88E+02±3.79E+00
26	2.05E+02±2.59E+01	**2.00E+02±1.85E-03**	**2.00E+02±2.31E-03**	**2.00E+02±1.83E-03**	**2.00E+02±2.98E-03**
27	**5.53E+02±2.04E+02**	6.30E+02±2.43E+02	6.22E+02±1.83E+02	5.86E+02±2.12E+02	7.72E+02±1.93E+02
28	**3.00E+02±0.00E+00**	**3.00E+02±0.00E+00**	**3.00E+02±0.00E+00**	**3.00E+02±0.00E+00**	**3.00E+02±0.00E+00**

Table 3. Results of 50D benchmark functions with different values of X

problem	AGDE	X = 1	X = [1-currFE/maxFE]	X = 0.5	X = 2
1	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
2	**1.27E+05±6.11E+04**	2.73E+05±8.80E+04	3.10E+05±1.16E+05	2.28E+05±8.90E+04	4.07E+05±1.58E+04
3	3.22E+06±7.74E+06	1.59E+05±1.90E+05	5.36E+05±6.27E+05	5.39E+05±8.77E+05	**1.41E+05±2.06E+05**
4	**4.40E+00±4.25E+00**	3.51E+01±2.74E+01	1.70E+01±1.06E+01	1.02E+01±7.89E+01	8.23E+01±5.15E+01
5	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
6	**4.34E+01±0.00E+00**	**4.34E+01±0.00E+00**	**4.34E+01±0.00E+00**	**4.34E+01±0.00E+00**	**4.34E+01±0.00E+00**
7	1.86E+01±8.43E+00	6.71E+01±3.35E+00	8.51E+01±3.95E+00	8.09E+01±3.89E+00	**7.85E+00±3.86E+00**
8	**2.10E+01±3.26E-02**	2.11E+01±3.38E-02	2.11E+01±3.78E-02	2.11E+01±3.66E-02	2.11E+01±3.53E-02
9	**5.06E+01±1.18E+01**	5.72E+01±2.27E+00	5.35E+01±4.19E+00	5.49E+01±4.54E+00	5.86E+01±1.78E+00
10	6.12E-02±2.99E-02	2.61E-02±1.91E-02	3.97E-02±2.34E-02	2.97E-02±1.84E-02	**2.46E-02±1.18E-02**
11	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
12	**1.96E+02±6.58E+01**	2.49E+02±5.79E+01	2.22E+02±3.80E+01	2.19E+02±5.06E+01	2.89E+02±2.44E+01
13	2.62E+02±4.98E+01	3.06E+02±2.34E+01	**2.59E+02±4.50E+01**	2.70E+02±3.53E+01	3.28E+02±2.13E+01
14	3.81E+02±6.21E+01	4.66E+02±6.58E+01	**2.24E+02±4.91E+01**	2.84E+02±5.20E+01	7.19E+02±1.09E+02
15	1.29E+04±3.81E+02	1.29E+04±4.65E+02	**1.25E+04±5.83E+02**	1.27E+04±4.14E+02	1.31E+04±3.30E+02
16	3.20E+00±3.78E-01	3.35E+00±2.31E-01	**3.18E+00±3.07E-01**	3.24E+00±3.08E-01	3.29E+00±2.86E-01
17	7.39E+01±2.07E+00	7.69E+01±2.62E+00	**6.41E+01±1.88E+00**	6.80E+01±2.48E+00	8.76E+01±2.80E+00
18	3.82E+02±1.56E+01	3.99E+02±1.78E+01	**3.74E+02±1.93E+01**	3.86E+02±1.44E+01	4.10E+02±1.79E+01
19	8.53E+00±6.01E-01	9.16E+00±4.76E-01	**7.90E+00±5.34E-01**	8.20E+00±5.29E-01	9.90E+00±6.32E-01
20	**2.16E+01±2.65E-01**	2.18E+01±2.64E-01	**2.16E+01±2.92E-01**	**2.16E+01±3.15E-01**	2.19E+01±3.40E-01
21	5.89E+02±4.27E+02	4.54E+02±4.03E+02	3.58E+02±3.30E+02	3.76E+02±3.46E+02	**3.08E+02±3.00E+02**
22	2.48E+03±2.42E+02	2.39E+03±5.98E+01	**3.27E+01±1.30E+01**	4.93E+01±2.68E+01	6.88E+02±1.27E+02
23	1.31E+04±5.37E+02	1.29E+04±4.78E+02	**1.25E+04±5.40E+02**	1.29E+04±4.80E+02	1.33E+04±4.54E+02
24	2.39E+02±1.32E+01	**2.19E+02±1.27E+01**	2.27E+02±1.26E+01	2.24E+02±1.49E+01	2.31E+02±2.00E+01
25	**3.60E+02±2.41E+01**	3.75E+02±5.56E+00	3.69E+02±7.09E+00	3.68E+02±1.26E+01	3.78E+02±5.57E+00
26	2.91E+02±9.68E+01	2.37E+02±8.63E+01	2.49E+02±8.98E+01	**2.32E+02±7.56E+01**	2.37E+02±8.60E+01
27	**1.10E+03±3.39E+02**	1.37E+03±3.26E+02	1.37E+03±2.79E+02	1.24E+03±2.98E+02	1.61E+03±2.25E+02
28	4.58E+02±4.11E+02	**4.00E+02±0.00E+00**	**4.00E+02±0.00E+00**	**4.00E+02±0.00E+00**	**4.00E+02±0.00E+00**

Statistical analysis is reported in order to compare the quality of solutions [37], two non-parametric statistical hypothesis tests are used to compare the results: (1) the Friedman test; and (2) Wilcoxon test with 0.05 significance level. The results are displayed in Tables 4 and 5. From Table 4, It is obvious that AGDE with non-linear reduction with the power (X = 1 − currFE/maxFE) is the best choice over the all dimensions with least ranking among all AGDE versions with other powers, as the quality of solutions provided by proposed power *X* is much better than the results provided by other powers. Besides, in Table 5, R+ is the sum of the ranks for AGDE, R− is the sum of the ranks for the EAGDE in each row, P-VALUE less than 0.05 indicates the superiority of EAGDE over the compared algorithm and Sig. is + if the EAGDE outperforms the compared algorithm and ≈ if there is no significant difference. It can be obviously seen from Table 5 that EAGDE obtains higher R + values than R– in most of the cases while slightly lower R+ value than R− value in comparison with AGDE with D = 30 and 50 with powers 1 and 2.

Table 4. Average ranks for 10D, 30D and 50D and different powers of *X*

Algorithm	AGDE	X = 1	X = 1 − *currFE*/*maxFE*	X = 0.5	X = 2
D = 10	4.21	2.79	**2.38**	2.79	2.84
D = 30	**2.61**	3.2	2.63	2.54	4.04
D = 50	2.84	3.27	**2.46**	2.5	3.93
Mean rank	3.22	3.08666667	**2.49**	2.61	3.603333

Table 5. Wilcoxon's test between AGDE and **EAGDE** with different powers of X for D = 10, 30 and 50, respectively

AGDE vs.	D = 10			D = 30			D = 50		
	R+	R−	P-value	R+	R−	P-value	R+	R−	P-value
X = 1	46.5	229.5	**0.005**	216.5	134.5	0.298	174	151	0.757
X = 1 − *currFE*/*maxFE*	29	247	**0.001**	108.5	216.5	0.146	96	229	0.074
X = 0.5	48.5	251.5	**0.004**	150.5	174.5	0.747	116	209	0.211
X = 2	22	231	**0.001**	245	106	0.077	228.5	122.5	0.178

At the end of this section, we can briefly mention the enhancements for the **AGDE** [34]:

1. The initial population size is 10D, 6D and 4D for the dimensions 10, 30 and 50 respectively.
2. The non-linear reduction formula in Eq. 2 with the power ***X*** = (1 − *currFE*/*maxFE*).

3 Evaluation of EAGDE on CEC'2013 Benchmark Functions

This section evaluates the performance of EAGDE on CEC'2013 benchmark functions [36]. CEC'2013 consists of 28 functions with different characteristics. Functions *f*1–*f*5 are unimodal functions, *f*6–*f*20 are basic multimodal functions and *f*21–*f*28 are composite functions.

3.1 Parameters Setup

Initial population size ($NP_{initial}$ was set to 10D, 6D and 4D for the 10, 30 and 50 dimensions respectively, mutation factor (F) is a uniform random number between 0.1 and 0.9, partition size (p) is 0.1, maximum number of function calls (*maxFE*) is set to 10,000 * D and N_{min} is set to 12. 51 independent runs for each problem is performed. An important guideline was followed that concerned with the gap between the known optimal solution and the best solution obtained from the algorithm, the solution is set to zero if the gap is greater than 10^{-8} [38].

3.2 Comparison Against EAGDE on CEC'2013 Benchmark Functions

EAGDE is compared with the AGDE and five state-of-art optimization algorithms: "The super-fit multicriteria adaptive differential evolution (SMADE), covariance matrix adaptive evolution strategy (CMAES), the self-adaptive Differential Evolution with PBX crossover (MDE_PBX), the Cooperatively Coevolving Particle Swarms Optimizers (CCPSO2) and Creatively-oriented optimization algorithm (COOA)" [34]. The statistical results of EAGDE including the best and standard deviation are displayed in Tables 1, 2 and 3. Due to space constraints, the reader can check the results of AGDE and the other five state-of-the-art algorithms in [34]. The statistical results showed that the EAGDE improved the solution quality.

To compare the quality of solutions from a statistical angle, two non-parametric statistical tests are used to compare the results: Friedman test and Wilcoxon sum rank test with p-value = 0.05. The results are displayed in Tables 6 and 7.

Table 6 shows that superiority of EAGDE over compared algorithms in solving 10D problems and is comparable with other algorithms in 30D and 50D problems. Overall, the EAGDE ranked second after SMADE but EAGDE is simpler in the structure than SMADE and has fewer control parameters, which proves the efficiency of the innovative ideas proposed in EAGDE.

Table 6. Average ranks for 10D, 30D and 50D

Algorithm	EAGDE	AGDE	COOA	SMADE	MDE-pBX	CMAES	CCPSO2
D = 10	**2.63E+00**	3.54E+00	3.45E+00	3.30E+00	4.32E+00	5.46E+00	5.30E+00
D = 30	3.63E+00	3.50E+00	3.38E+00	**3.27E+00**	4.46E+00	5.23E+00	4.54E+00
D = 50	3.88E+00	4.00E+00	3.36E+00	**3.14E+00**	4.34E+00	4.91E+00	4.38E+00
Mean rank	3.38E+00	3.68E+00	3.40E+00	**3.24E+00**	4.37E+00	5.20E+00	4.74E+00

Table 7. Wilcoxon's test between EAGDE, AGDE and the state-of-the-art algorithms optimization algorithm

EAGDE vs.	10D				30D				50D			
	R+	R−	P-value	Sig.	R+	R−	P-value	Sig.	R+	R−	P-value	Sig.
AGDE	29	247	**0.001**	+	108.5	216.5	0.146	≈	96	229	0.074	≈
COOA	155	223	0.414	≈	148	203	0.485	≈	176	175	0.990	≈
SMADE	61	239	**0.011**	+	182	224	0.633	≈	216	162	0.517	≈
MDE_*p*BX	32	293	**0.000**	+	116	290	**0.048**	+	126	252	0.130	≈
CMAES	25	300	**0.000**	+	87	291	**0.014**	+	130	248	0.156	≈
CCPSO2	14	311	**0.000**	+	100	278	**0.032**	+	125	253	0.124	≈

In Table 7, R+ is the sum of the ranks for EAGDE, R− is the sum of the ranks for the compared algorithm in each row, P-VALUE less than 0.05 indicates the superiority of EAGDE over the compared algorithm and Sig. is + if the EAGDE outperforms the compared algorithm and ≈ if there is no significant difference. It is obvious from Table 7 that EAGDE outperforms 4 and 2 algorithms in 10D and 30D, respectively, and shows no significant difference in other algorithms.

4 Conclusion and Future Work

Population size plays a vital rule in the evolution process of the DE. Large population size increases the exploration but consumes more resources, while small population size may lead to local optima, premature convergence and/or stagnation. This paper introduced a new rule for selecting the initial population size that is related to the problem dimensionality, and a new non-linear population reduction formula in order to enhance the performance of AGDE algorithm proposed by Mohamed and Mohamed [34]. The main idea behind designing population reduction technique is based on deleting the worst solutions greatly during the early-middle stages of the optimization process as well as keep the promising solutions during the middle - later stages of the optimization process as long as possible. The new idea proved superiority and novelty of EAGDE when applied on CEC'2013 benchmark functions and the results were compared with AGDE and other state-of-the-art optimization algorithms. The quality of the solution is improved. Additionally, future research will investigate the performance of the EAGDE algorithm in solving unconstrained and constrained multi-objective optimization problems as well as real-world applications.

References

1. Storn, R., Price, K.: Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces. International Computer Science Institute Technical Report, Technical report. TR-95-012 (1995)
2. Storn, R., Price, K.: Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. J. Global Optim. **11**(4), 341–359 (1997)

3. Das, S., Abraham, A., Chakraborty, U.K., Konar, A.: Differential evolution using a neighborhood-based mutation operator. IEEE Trans. Evol. Comput. **13**(3), 526–553 (2009)
4. Zhang, J., Sanderson, A.C.: JADE: adaptive differential evolution with optional external archive. IEEE Trans. Evol. Comput. **13**(5), 945–958 (2009)
5. Qin, A.K., Huang, V.L., Suganthan, P.N.: Differential evolution algorithm with strategy adaptation for global numerical optimization. IEEE Trans. Evol. Comput. **13**(2), 398–417 (2009)
6. Mohamed, A.W., Sabry, H.Z.: Constrained optimization based on modified differential evolution algorithm. Information Sciences, pp. 171–208 (2012)
7. Mohamed, A.W., Sabry, H.Z., Khorshid, M.: An alternative differential evolution algorithm for global optimization. J. Adv. Res. **3**(2), 149–165 (2012)
8. Mohamed, A.W., Sabry, H.Z., Farhat, A.: Advanced differential evolution algorithm for global numerical optimization. In: Proceedings of the IEEE International Conference on Computer Applications and Industrial Electronics (ICCAIE 2011), Penang, Malaysia, pp. 156–161 (2011)
9. Li, X., Yin, M.: Modified differential evolution with self-adaptive parameters method. J. Comb. Optim. **31**(2), 546–576 (2014)
10. Mohamed, A.W.: An improved differential evolution algorithm with triangular mutation for global numerical optimization. Comput. Ind. Eng. **85**, 359–375 (2015)
11. Mohamed, A.W., Suganthan, P.N.: Real-parameter unconstrained optimization based on enhanced fitness-adaptive differential evolution algorithm with novel mutation. Soft. Comput. (2017). https://doi.org/10.1007/s00500-017-2777-2
12. Mohamed, A.W.: An efficient modified differential evolution algorithm for solving constrained non-linear integer and mixed-integer global optimization problems. Int. J. Mach. Learn. Cybernet. **8**, 989 (2017). https://doi.org/10.1007/s13042-015-0479-6
13. Mohamed, A.W.: A novel differential evolution algorithm for solving constrained engineering optimization problems. J. Intell. Manuf. (2017). https://doi.org/10.1007/s10845-017-1294-6
14. Mohamed, A.W., Almazyad, A.S.: Differential evolution with novel mutation and adaptive crossover strategies for solving large scale global optimization problems. Appl. Comput. Intell. Soft Comput. **2017**, 18 (2017). https://doi.org/10.1155/2017/7974218
15. Mohamed, A.W.: Solving stochastic programming problems using new approach to Differential Evolution algorithm. Egypt. Inform. J. **18**(2), 75–86 (2017)
16. Brest, J., Greiner, S., Boškovic, M., Mernik, M., ˇZumer, V.: Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. IEEE Trans. Evol. Comput. **10**(6), 646–657 (2006)
17. Noman, N., Iba, H.: Accelerating differential evolution using an adaptive local search. IEEE Trans. Evol. Comput. **12**(1), 107–125 (2008)
18. Peng, F., Tang, K., Chen, G., Yao, X.: Multi-start JADE with knowledge transfer for numerical optimization. In: IEEE CEC, pp. 1889–1895 (2009)
19. Montgomery, J., Chen, S.: An analysis of the operation of differential evolution at high and low crossover rates. In: IEEE Congress on Evolutionary Computation, Barcelona, pp. 1–8 (2010)
20. Mallipeddi, R., Suganthan, P.N., Pan, Q.K., Tasgetiren, M.F.: Differential evolution algorithm with ensemble of parameters and mutation strategies. Appl. Soft Comput. **11**(2), 1679–1696 (2011)
21. Wang, Y., Cai, Z., Zhang, Q.: Differential evolution with composite trial vector generation strategies and control parameters. IEEE Trans. Evol. Comput. **15**(1), 55–66 (2011)

22. Yong, W., Han-Xiong, L., Tingwen, H., Long, L.: Differential evolution based on covariance matrix learning and bimodal distribution parameter setting. Appl. Soft Comput. **18**, 232–247 (2014)
23. Draa, A., Bouzoubia, S., Boukhalfa, I.: A sinusoidal differential evolution algorithm for numerical optimization. Appl. Soft Comput. **27**, 99–126 (2015)
24. Das, S., Suganthan, P.N.: Differential evolution: a survey of the state-of-the-art. IEEE Trans. Evol. Comput. **15**(1), 4–31 (2011)
25. Das, S., Mullick, S.S., Suganthan, P.N.: Recent advances in differential evolution-an updated survey. Swarm Evol. Comput. **27**, 1–30 (2016)
26. Cheng, J.X., Zhang, G.X., Neri, F.: Enhancing distributed Differential Evolution with multicultural migration for global numerical optimization. Inf. Sci. **247**, 72–93 (2013)
27. Gao, W.F., Pan, Z., Gao, J.: A new highly efficient differential evolution with self-adaptive strategy for multimodal optimization. IEEE Trans. Cybern. **44**(8), 1314–1327 (2014)
28. Mallipeddi, R., Suganthan, P.N.: Empirical study on the effect of population size on Differential Evolution algorithm. In: Proceedings of IEEE Congress on Evolutionary Computation, Hong Kong (2008)
29. Wang, H., Wang, W.J., Cui, Z.H., Sun, H., Ranhnamayan, S.: Heterogeneous differential evolution for numerical optimization. Sci. World J. **2014**, 7 pages (2014). Article no. 318063, https://doi.org/10.1155/2014/318063
30. Gao, W.F., Yen, G.G., Liu, S.Y.: A dual Differential Evolution with coevolution for constrained optimization. IEEE Trans. Cybern. **45**(5), 1094–1107 (2015)
31. Brest, J., Maucec, M.S.: Self-adaptive Differential Evolution algorithm using population size reduction and three strategies. Soft. Comput. **15**(11), 2157–2174 (2011)
32. Zamuda, A., Brest, J.: Self-adaptive control parameters' randomization frequency and propagations in Differential Evolution. Swarm Evol. Comput. **25**, 72–99 (2015)
33. Piotrowski, A.P.: Review of differential evolution population size. Swarm Evol. Comput. **32**, 1–24 (2017)
34. Mohamed, A.W., Mohamed, A.K.: Adaptive guided differential evolution algorithm with novel mutation for numerical optimization. Int. J. Mach. Learn. Cyber. (2017). https://doi.org/10.1007/s13042-017-0711-7
35. Laredo, J.L.J., Fernandes, C., Guervós, J.J.M., Gagné, C.: Improving genetic algorithms performance via deterministic population shrinkage. In: GECCO 2009, pp. 819–826 (2009)
36. Liang, J.J., Qin, B.Y., Suganthan, P.N., Hernandez-Diaz, A.G.: Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization, Zhengzhou University, Nanyang Technological University, Zhengzhou, China, Singapore (2013)
37. García, S., Molina, D., Lozano, M., Herrera, F.: A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behavior: a case study on the CEC'2005 special session on real parameter optimization. J. Heuristics **15**, 617–644 (2009)
38. Hansen, N., Ostermeier, A.: CMA-ES source code (2009). http://www.lri.fr/~hansen/cmaes_inmatlab.html

Simulated Annealing Based Quantum Inspired Automatic Clustering Technique

Alokananda Dey[1], Sandip Dey[2], Siddhartha Bhattacharyya[1](✉), Vaclav Snasel[3], and Aboul Ella Hassanien[4]

[1] Department of Computer Application,
RCC Institute of Information Technology,
Beliaghata, Kolkata 700015, India
alokananda_22@yahoo.co.in, dr.siddhartha.bhattacharya@gmail.com

[2] Department of Computer Science and Engineering,
OmDayal Group of Institutions,
Birshibpur, Howrah 711316, India
dr.ssandip.dey@gmail.com

[3] Faculty of Electrical Engineering and Computer Science,
VSB-Technical University of Ostrava,
Ostrava, Czech Republic
vaclav.snasel@vsb.cz

[4] Information Technology Department,
Faculty of Computers and Information,
Cairo University, Giza, Egypt
aboitcairo@gmail.com

Abstract. Cluster analysis is a popular technique whose aim is to segregate a set of data points into groups, called clusters. Simulated Annealing (SA) is a popular meta-heuristic inspired by the annealing process used in metallurgy, useful in solving complex optimization problems. In this paper, the use of the Quantum Computing (QC) and SA is explored to design Quantum Inspired Simulated Annealing technique, which can be applied to compute optimum number of clusters for image clustering. Experimental results over a number of images endorse the effectiveness of the proposed technique pertaining to *fitness value*, *convergence time*, *accuracy*, *robustness*, and *standard error*. The paper also reports the computation results of a statistical superiority test, known as *t*-test. An experimental judgement to the classical technique has also be presented, which eventually demonstrates that the proposed technique outperforms the other.

Keywords: Simulated annealing · Automatic clustering
Cluster validity · Quantum computing

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 73–81, 2018.
https://doi.org/10.1007/978-3-319-74690-6_8

1 Introduction

Clustering or cluster analysis [1,2], an unsupervised learning method, can be described as a grouping of similar data points in consonance with a selected similarity metric. A cluster is thus an assortment of objects which possess similarity between themselves and dissimilarity to the other objects associated with other clusters [1–3]. The objective of the different clustering algorithms is also to find a natural framework or relationship among data points, especially in unlabeled data set. Since the last few decades, cluster analysis has significantly become a popular research area of interest in several application domains, such as engineering, social sciences, business and many others. There exist several clustering algorithms in the literature. So far, some clustering algorithms have been applied successfully and effectively in various domains, which may include text mining [4], object recognition [5] and intrusion detection [6], to name a few. At times, clustering of data and evaluating its quality become two important aspects in this field. Several clustering algorithms can be used to determine the number of clusters in any given data set, where the quality of clustering is usually measured using different validity indices [7,8].

In the twenty-first century, quantum computation is probably the foremost challenging task for computer science and engineering and many others [9–11]. A quantum computing is a mechanism that can be applied to perform calculations by concepts of quantum mechanics, in which the behavior of particles is accounted at the atomic/sub-atomic level of computation. Quantum computer can perform millions of processing at a time. So on and so forth, efforts have been made to design quantum version of traditional algorithms to enhance their processing capability. These quantum inspired algorithms induce the QCs phenomena like superposition, entanglement, etc., which in turn help them to outperform classical algorithms to a great extent [12,13]. Some of these algorithms are available in the literature [14–17].

These days, meta-heuristic methods are extensively used in numerous domains of science and engineering. These techniques can be most suitably described as the popular algorithmic framework, which is probably the most useful techniques to unearth the appropriate solutions of different combinatorial optimization problems. They can be successfully applied in various problem domains with comparatively no or few adjustments using some strategic guidelines. Meta-heuristics sometimes gather information from other sources to direct searching mechanism towards the global optima. They explore their search space to determine optimal solution within the stipulated time. Meta-heuristics are equally effective to solve both simple and complex problems. A few typical applications of these techniques are given in [18,19].

In the past few years, tremendous efforts have been made by different scholars in clustering through evolutionary techniques. In this paper, a new technique, called Quantum Inspired Simulated Annealing Based Automatic Clustering technique, has been introduced. This technique can be effectively applied on different gray scale [20] and Berkeley images (Benchmark dataset) [21] to find optimal number of clusters on the run.

2 Preliminaries

This section provides a brief explanation of the basic framework of simulated annealing and quantum computing along with some of the key basic concepts.

2.1 Simulated Annealing

Simulated annealing is a well-known heuristic-search optimization technique, introduced by Kirkpatrik *et al.* [22] in 1983. SA emulates the annealing procedure used in metallurgy, which is the branch of material science dealing with metals and alloys. In general, when a material experiences the process of annealing, firstly, the material is heated until it attains its fusion point to deliquesce it, and then gradually cooled down in a regulated way until it solidifies back. SA exploits the concept of annealing to seek optimal solutions to the concern combinatorial optimization and other related problems. This technique searches for solutions by beginning at a very high "temperature" and primarily admits all suggested arrangements. As the temperature cuts down, it turns into more discriminating, and the part of shifts which causes worsening the solution is brought down. If the cooling process is done in a corrective way, this unearths an arrangement which is very close or equivalent to the crystal structure, resulting optimal or near-optimal solutions.

The foremost advantage of SA over other meta-heuristics is that it can escape trapping from the local minima. When a random searching is initiated by this technique, it not only agrees to make changes that decrease the fitness function, say g (for a minimization problem) but also accepts changes that increase g (for a maximization problem). The latter can be accepted with certain acceptance probability, as given by the following Equation.

$$p = \exp\left[\frac{\Delta g}{T}\right] \tag{1}$$

where, Δg represents the difference between the two fitness values of g in successive iterations, and T is called control parameter. It has been proved that the SA can reach at global optimum if the rate of cooling is controlled in a proper way.

2.2 Quantum Computing

A quantum bit is a quantum system where boolean states ($0\,\&\,1$) are described as a prearranged pair of normalized, mutually orthogonal quantum states, as denoted by $\{|0\rangle, |1\rangle\}$, where, $|0\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ and $|1\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$. These two states are the basic vector states, which cause the foundation of a computational basis in two-dimensional, complex vector space $\mathbb{C}^2$. Other state of the quantum bit can be expressed as a superposition $|\psi\rangle = \alpha|0\rangle + \beta|0\rangle$, where, $(\alpha, \beta \in \mathbb{C})$ and $\alpha^2 + \beta^2 = 1$ [23].

A quantum register of a quantum mechanical system is analogous to a classical register of its classical counterpart. A group of m quantum bits is collectively known to be a quantum register of size m. Mathematically, tensor products of a quantum bit $< bra > / < ket >$ vectors are used to achieve a quantum register. For example, suppose, $|b\rangle$ represents the tensor product $|b_0\rangle \bigotimes |b_1\rangle \bigotimes |b_2\rangle \bigotimes \cdots \bigotimes |b_{m-1}\rangle$, where, $b_j \in \{0, 1\}$ and $j \in [0, 1, \ldots, (m - 1)]$. Hence, numbers between $[0, 2m]$ is represented by $2m$ states. Therefore, a quantum register of size four can hold numbers like 10 or 12 as $|1\rangle \bigotimes |0\rangle \bigotimes |1\rangle \bigotimes |0\rangle$ and $|1\rangle \bigotimes |1\rangle \bigotimes |0\rangle \bigotimes |0\rangle$ $|1\rangle \bigotimes |0\rangle|1\rangle \bigotimes |0\rangle$ or $|1\rangle \bigotimes |1\rangle|0\rangle \bigotimes |0\rangle$, respectively [14–17].

A quantum logic gate or in short qgate is a device which can be used to carry out a fixed unitary operation on chosen quantum bits over a fixed period. Several quantum logic gates can be coalesced to construct a device, called quantum network, where the computational steps of these qgates are synchronized in time. The network size depends on the number of gates it has. Some examples of the quantum logic gate are Fredkin gate, NOT gate, Toffoli gate, C-NOT gate, controlled phase-shift gate, Hadamard gate, etc. Quantum entanglement is the correlation between a pair of quantum systems or objects defined by quantum mechanics [9–11].

3 Clustering Validation Techniques and Fitness Function

The purpose of cluster analysis is to identify and collate similar objects into groups and, thus help to find pattern distribution and intriguing correlations in a wide range of datasets. Eventually, in the last few years, the use of the huge volume of experimental data sets, accelerated the need for implementing clustering algorithms on a large scale in various domains. Hence, evaluating the "goodness" of resulting clusters has appeared to be an important issue in this regard. The quality of clustering can be measured using cluster validity indices. There are some popular cluster validity indices present in the literature. These matrices usually assess the quality described above using an iterative approach [7,8]. Two important aspects of the cluster validity index are compactness and separation. Compactness describes the closeness of the members of each existing cluster. The objects of a cluster should remain as close as possible to other objects. Variance is a popular measure of compactness, which should be minimized. For separation, the distance between different clusters should be as maximum as possible [7,8].

In the proposed methodology, Davies-Bouldin index [24] has been used as the fitness function for experimental purpose. The target is to find the sets of clusters which are compact and simultaneously well separated. The Davies-Bouldin index is formally defined as follows:

$$DB = \frac{1}{n} \sum_{j=1, j \neq k}^{n} \max \left[\frac{\delta_j + \delta_k}{d(c_j, c_k)} \right] \tag{2}$$

where, n denotes the number of clusters, δ_j and δ_k are the average distances of each data point in cluster j and k to their respective cluster centers c_j and c_k, respectively. Minimum values of DB indicates better quality of clustering, that means, clusters are compact, and the distance between any two centers are very large. Consequently, the optimal number of clusters can be obtained by minimizing the DB values, as mentioned in Eq. (2).

4 The Proposed Automatic Clustering Technique

A new technique for automatic clustering (QIACSA) of real life gray scale images [20] and Berkeley images (Benchmark dataset) [21] has been implemented through a quantum-inspired clustering algorithm. A popular clustering validity index, called Davies-Bouldin index (DB) [24] has been applied as an objective function for assessing the performance of the proposed technique. At initial, a configuration, $\mathscr{P}$ of length $\mathscr{L}$ is created with real random numbers between $(0, 1)$. r number of cluster centers are selected from $\mathscr{P}$ at random. After that, the element $\mathscr{P}$ is encoded between $(0, 1)$ using an encoding scheme, which produces $\mathscr{P}^{+}$. Afterward, a fundamental feature of QC, called *quantum orthogonality* is applied in $\mathscr{P}^{+}$ to create $\mathscr{P}^{++}$. Then a cluster validity index, known as Davies-Bouldin index [24] is introduced as the objective function to measure the fitness of the configuration. Let it be stored in E_{curr}. At first, an extremely high temperature, T_{init} is set for a temperature parameter, Tt. For each temperature, the proposed technique is executed for the ι number of iterations. Thereafter, the value of Tt is brought down by $Tt = T_{init} * \exp(-(c1 * t)/d)$, where, $c1 \in (0, 1)$ is constant, t denotes the iteration number at particular point of time and d is the dimension of the input space. This procedure is continued until Tt reaches the predefined culminating temperature, T_{fnl}. At each value of ι, the element of $\mathscr{P}$ is perturbed at several points at random, expecting to have a better configuration than the previous one. Let it be named $\mathscr{S}$.

The newly created configuration is accepted based on the condition given by $E_{next} < E_{curr}$, where, E_{next} is the fitness value of $\mathscr{S}$. If $E_{next} \geq E_{curr}$, this configuration can also be accepted with a probability $\exp(-(E_{curr} - E_{next}))/Tt$, which is defined by the Boltzmann distribution. Note that, the values of $c1$ and d are selected as 0.7 and 1.0, respectively. The outline of QIACSA is depicted in Algorithm 1.

The time complexity analysis (worst case) of QIACSA is presented in the following steps.

- Step 1: Since $\mathscr{P}$ contains a single configuration, the time complexity becomes $O(\mathscr{L})$, where $\mathscr{L}$ denotes the length of the configuration.

- Steps 2–4: Like step 1, these three consecutive steps perform identical number of computations. So, the time complexity for evaluating these steps become $O(\mathscr{L})$.
- Step 5: Normalizing data set leads to the time complexity to become $O(l_n)$, where, l_n is the size of the data set.
- Steps 7–19: Let the outer 'while loop' and inner 'for loop' are executed ϖ and ι number of times respectively in the proposed technique. Hence, the time complexity for executing these steps in QIACSA turns out to be $O(\varpi \times \iota)$.

So, aggregating the overall discussion, the QIACSA possesses the time complexity (worst case) $O(\mathscr{L} \times \varpi \times \iota)$.

Algorithm 1. Steps of QIACSA

```
1: Initialize the configuration containing $r$ number of randomly chosen cluster centers of population $\mathscr{P}$. Let each configuration in $\mathscr{P}$ has a length $\mathscr{L} = \lfloor\sqrt{gv}\rfloor$, where $gv$ stands for maximum pixel intensity value of the test image (gray scale) and $r \in [3, \mathscr{L}]$.
2: Encode the element in $\mathscr{P}$ to a real number between (0,1). Let it produces $\mathscr{P}^+$.
3: Apply the basic feature of QC, called *quantum orthogonality* to each element in $\mathscr{P}$. Let it produces $\mathscr{P}^{++}$.
4: Use Eq. (2) to compute fitness value of the configuration in $\mathscr{P}^{++}$. Let it be recorded in $E_{curr}$.
5: Normalize the input dataset (pixel intensity values of test image) between (0,1). Let it be called $\mathscr{D}$.
6: Set $Tt = T_{init}$ and $t = 1$.
7: while $Tt > T_{fnl}$ do
8:     for $i = 1$ to $\iota$ do
9:         A new configuration, called $\mathscr{S}$ is being created by perturbing the population, $\mathscr{P}$.
10:        Repeat steps (2) and (3) to produce $\mathscr{S}^+$ and $\mathscr{S}^{++}$.
11:        Evaluate the fitness of $\mathscr{S}^{++}$ using Eq. (2). Let it be denoted by $E_{next}$.
12:        if $E_{next} < E_{curr}$ then
13:            Set $\mathscr{P} = \mathscr{S}$, $\mathscr{P}^{++} = \mathscr{S}^{++}$ and $E_{curr} = E_{next}$.
14:        else
15:            Set $\mathscr{P} = \mathscr{S}$, $\mathscr{P}^{++} = \mathscr{S}^{++}$ and $E_{curr} = E_{next}$ with the probability $\exp(-(E_{curr} - E_{next}))/Tt$.
16:            Set $t = t + 1$ and $Tt = T_{init} * \exp(-(c1 * t)/d)$, where, $c1 \in (0, 1)$ is constant and $d$ is the dimension of the input space.
17:        end if
18:    end for
19: end while
20: Report the optimal number of cluster, $n_c$ and the corresponding fitness value, $DB$.
```

5 Experimental Results and Analysis

The proposed technique has been implemented in Python environment. It has been tested over three real-life gray scale images [20] and three other Berkeley images [21], as part of the experimental purpose. The selected real life images are **Couple**, **Clown** and **Kiel** [20] of size 512 × 512 each. The Berkeley images [21] are ♯87046, ♯89072 and ♯86016 having dimensions 481 × 321 and 80 × 120 and 120 × 80, respectively. Typical execution times on an AMD Dual Core Processor C60 with Turbo Core Technology upto 1.333 GHz, 4 GB DDR3 using Windows 7 environment have been presented in Table 1. Experiments have been conducted 20 times for each data set. The performances of QIACSA and its classical counterpart have been evaluated and compared by following issues. (a) Results of automatic clustering, (b) Analyzing stability and accuracy, (c) Standard error computation, (d) Computational time measurement (in seconds), and (e) Results of unpaired t-tests.

The experimental results yielded by each technique are grouped in Table 1. It contains the results of clustering, such as optimal number of clusters along with the fitness values obtained from both techniques for each test image. Through these results, it can be clearly observed that the proposed technique provides a better clustering than its counterpart for all cases.

Table 1. Best results for QIACSA and ACSA

QIACSA																	
Couple			**Clown**			**Kiel**			♯87046			♯89072			♯86016		
n_c	DB	t	n_c	DB	t	n_c	DB	t	n_c	DB	t	n_c	DB	t	n_c	DB	t
4	0.036	637	6	0.072	772	4	0.072	913	4	0.088	1642	5	0.072	551	4	0.048	102
ACSA																	
Couple			**Clown**			**Kiel**			♯87046			♯89072			♯86016		
n_c	DB	t	n_c	DB	t	n_c	DB	t	n_c	DB	t	n_c	DB	t	n_c	DB	t
3	0.136	1643	6	0.075	870	4	0.216	978	3	0.194	2775	4	0.201	667	4	0.122	363

To evaluate the performance of QIACSA, it has been implemented on variety of images. The effectiveness of the proposed technique has been examined by choosing a group of performance metrics like computational time to get an idea of time complexity; mean and standard deviation to determine the accuracy and stability of the proposed technique; standard error to estimate the standard deviation of fitness values associated with the proposed technique; unpaired t-test to establish the statistical superiority of QIACSA over other comparable technique.

Table 2 shows the mean (μ), standard deviation (σ) and standard error (ε) associated with the results of different techniques over all runs. These results indicate that QIACSA is better than the other technique with regard to accuracy, stability and standard deviation of the error. From Table 1, it is clearly evident that, in all cases, the proposed technique consumes lesser computation time. Hence, the superiority of QIACSA is computationally established.

An unpaired t-test has been conducted between these two techniques at 5% significant level. This test yield p-value, which may vary varies from 0% to 100%. This test is performed to check whether the p-value obtained is less than 0.05 or

Table 2. Mean (μ), standard deviation (σ) and standard error (ε) computed for QIACSA and ACSA

QIACSA																	
Couple			**Clown**			**Kiel**			♯87046			♯89072			♯86016		
μ	σ	ε	μ	σ	ε	μ	σ	ε	μ	σ	ε	μ	σ	ε	μ	σ	ε
0.076	0.024	0.009	0.103	0.014	0.005	0.096	0.011	0.004	0.101	0.017	0.007	0.086	0.009	0.003	0.083	0.020	0.007
ACSA																	
Couple			**Clown**			**Kiel**			♯87046			♯89072			♯86016		
μ	σ	ε	μ	σ	ε	μ	σ	ε	μ	σ	ε	μ	σ	ε	μ	σ	ε
0.219	0.042	0.015	0.158	0.052	0.019	0.290	0.058	0.022	0.256	0.038	0.017	0.269	0.050	0.015	0.244	0.089	0.033

Table 3. Results of unpaired t-test (p-value) between QIACPSO and ACPSO

Data set	Couple	Clown	Kiel	♯87046	♯89072	♯86016
p-**value**	< 0.00005	0.0200	< 0.00001	0.00003	< 0.00001	0.0005
Significance level	1	2	1	1	1	1
1 :→ **Extremely significant** 2 :→ **Significant**						

not. If $p < 0.05$, it shows that the alternative hypothesis is accepted. The results of this test among the participating techniques have been reported in Table 3. The results proves that QIACPSO outperforms other.

6 Concluding Remarks

This paper introduces an automatic image clustering approach on the basis of quantum computing and simulated annealing. The clustering process has come up as an optimization problem. Experimental evidence proves that the proposed technique has an adequate conciliation between its *fitness value*, *accuracy*, *stability*, *convergence time*, *statistical accuracy of the estimation* and its *statistical superiority* as to other comparable technique.

However, techniques persists to be examined to apply the proposed technique to optimize the clustering of true color images. The authors are at present involved in this direction.

References

1. Jain, A., Dubes, R.: Algorithms for Clustering Data. Prentice Hall, Upper Saddle River (1988)
2. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: a review. ACM Comput. Surv. **31**(3), 264–323 (1999)
3. Chou, C.H., Su, M.C., La, E.: A new cluster validity measure and its application to image compression. Pattern Anal. Appl. **7**(2), 205–250 (2004)
4. SanJuan, E., Ibekwe-SanJuan, F.: Text mining without document context. Inf. Process. Manage. **42**(6), 1532–1552 (2006)

5. Perdisci, R., Giacinto, G., Roli, F.: Alarm clustering for intrusion detection systems in computer networks. Eng. Appl. Artif. Intell. **19**(4), 429–438 (2006)
6. Jaenichen, S., Perneri, P.: Acquisition of concept descriptions by conceptual clustering (2005)
7. Maulik, U., Bandyopadhyay, S.: Performance evaluation of some clustering algorithms and validity indices. IEEE PAMI **24**, 1650–1654 (2002)
8. Halkidi, M., Batistakis, Y., Vazirgiannis, M.: On clustering validation techniques. J. Intell. Inf. Syst. **17**(2), 107–145 (2001)
9. Dey, S., Bhattacharyya, S., Maulik, U.: Quantum inspired genetic algorithm and particle swarm optimization using chaotic map model based interference for gray level image thresholding. Swarm Evol. Comput. **15**, 38–57 (2014)
10. Dey, S., Bhattacharyya, S., Maulik, U.: Efficient quantum inspired meta-heuristics for multi-level true colour image thresholding. Appl. Soft Comput. **56**, 472–513 (2017)
11. Vendral, V., Plenio, M.B., Rippin, M.A.: Quantum entanglement. Phys. Rev. Lett. **78**(12), 2275–2279 (1997)
12. Dey, S., Saha, I., Bhattacharyya, S., Maulik, U.: Multi-level thresholding using quantum inspired meta-heuristics. Knowl.-Based Syst. **67**, 373–400 (2014)
13. Mcmohan, D.: Quantum Computing Explained. Wiley, Hoboken (2008)
14. Dey, S., Bhattacharyya, S., Maulik, U.: New quantum inspired meta-heuristic techniques for multi-level colour image thresholding. Appl. Soft Comput. **46**, 677–702 (2016)
15. Dey, S., Bhattacharyya, S., Maullik, U.: Quantum behaved swarm intelligent techniques for image analysis: a detailed survey. In: Bhattacharyya, S., Dutta, P. (eds.) Handbook of Research on Swarm Intelligence in Engineering. IGI Global, Hershey (2015)
16. Dey, S., Bhattacharyya, S., Maullik, U.: Optimum gray level image thresholding using a quantum inspired genetic algorithm. In: Advanced Research on Hybrid Intelligent Techniques and Applications (2015)
17. Han, K.H., Kim, J.H.: Quantum-inspired evolutionary algorithm for a class combinational optimization. IEEE Trans. Evol. Comput. **6**(6), 580–593 (2002)
18. Blum, C., Roli, A.: Metaheuristic in combinatorial optimization: overviewand conceptual comparison. Technical report, IRIDIA, 2001-13
19. Glover, F., Kochenberger, G.A.: Handbook on Metaheuristics. Kluwer Academic Publishers, New York (2003)
20. Real life gray scale images, Domain generated in September 2006. Accessed 26 Aug 2017
21. Benchmark dataset, Page generated Fri Oct 31 12:01:51 2003. Accessed 26 Aug 2017
22. Kirkpatrik, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. Science **220**, 671–680 (1983)
23. Dey, S., Bhattacharyya, S., Maulik, U.: Chaotic map model based interference employed in quantum inspired genetic algorithm to determine the optimum gray level image thresholding. In: Global Trends in Intelligent Computing Research and Development, pp. 68–110 (2013)
24. Davies, D., Bouldin, D.: A cluster separation measure. IEEE PAMI **1**(2), 224–227 (1979)

Hybrid Grasshopper Optimization Algorithm and Support Vector Machines for Automatic Seizure Detection in EEG Signals

Asmaa Hamad[1,3](✉), Essam H. Houssein[1,3],
Aboul Ella Hassanien[2,3], and Aly A. Fahmy[2]

[1] Faculty of Computers and Information, Minia University, Minya, Egypt
asmaa_hamad222@yahoo.com
[2] Faculty of Computers and Information, Cairo University, Giza, Egypt
[3] Scientific Research Group in Egypt (SRGE), Cairo, Egypt
http://www.egyptscience.net

Abstract. In this paper, a hybrid classification model using Grasshopper Optimization Algorithm (GOA) and support vector machines (SVMs) for automatic seizure detection in EEG is proposed called GOA-SVM approach. Various parameters were extracted and employed as the features to train the SVM with radial basis function (RBF) kernel function (SVM-RBF) classifiers. GOA was used for selecting the effective feature subset and the optimal settings of SVMs parameters in order to obtain a successful EEG classification. The experimental results confirmed that the proposed GOA-SVM approach, able to detect epileptic and could thus further enhance the diagnosis of epilepsy with accuracy 100% for normal subject data versus epileptic data. Furthermore, the proposed approach has been compared with Particle Swarm Optimization (PSO) with support vector machines (PSO-SVMs) and SVM using RBF kernel function. The computational results reveal that GOA-SVM approach achieved better classification accuracy outperforms both PSO-SVM and typical SVMs.

Keywords: EEG · Epilepsy · DWT · GOA · SVM

1 Introduction

Epilepsy is a kind of neurological disorder disease which is characterized by the recurrence of sudden abnormal reactions of the brain as epileptic seizures. About 40 or 50 million people in the world suffer from epilepsy [1]. So far, the specific cause of epilepsy in individuals is often unknown and the mechanisms behind seizure are little known. Electroencephalography (EEG) is the recording of the electrical activity of the brain, regularly taken through several electrodes at the scalp. EEG contains lots of worthy information relating to the numerous physiological states of the brain and thus is a very useful tool for understanding the brain diseases, such as epilepsy [2]. EEG signals of epileptic patients exhibit

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 82–91, 2018.
https://doi.org/10.1007/978-3-319-74690-6_9

two states of abnormal activities namely interictal or seizure free and ictal [3]. The interictal EEG signals are transitory waveforms and exhibit spikes, sharp or spiky waves. The ictal EEG signals are persistent waveforms with spikes and sharp wave complexes. Each EEG is ordinarily decomposed into five sub-bands: delta, theta, alpha, beta, and gamma [4].

Several algorithms have been developed in the literature to improve the detection and classification of epileptic EEG Signals. Authors in [5], developed a scheme for detecting epileptic seizures from EEG data recorded from epileptic patients and normal subjects. This scheme is based on DWT analysis and approximate entropy (ApEn) of EEG signals. SVM and (feed-forward backpropagation neural network) FBNN are used for classification purpose. Also in [6], authors proposed an epileptic seizure detection technique from brain EEG signals. Moreover, SVM and KNN are utilized for the classification process. Furthermore, in [7], authors proposed automated epileptic seizure detection that used permutation entropy (PE) as a feature. SVM is used to classify segments of normal and epileptic EEG based on PE values. The proposed system uses the fact that the EEG during epileptic seizures is described by PE than normal EEG.

This paper presents a method for selecting features that used as input data for classifiers and the best parameters for SVMs via applying an optimization algorithm [8,9]. SVMs have two types of parameters (penalty constant C parameter and kernel functions parameters), and the values of these parameters affect the performance of SVMs [9,10]. Selecting these parameters correctly guarantees to obtain the best classification accuracy [11]. Accordingly, this paper adopts GOA to present a novel GOA-SVM hybrid optimized classification system for epileptic EEG signals classification. The obtained experimental results obviously indicate significant enhancements in terms of classification accuracy achieved by the proposed GOA-SVMs classification system compared to classification accuracy achieved by the typical SVMs classification algorithm and Particle Swarm Optimization with SVM (PSO-SVM).

The rest of the paper is drawn as follows: Sect. 2 presents the materials and methods. The proposed classification approach is provided in Sect. 3. Experimental results and discussions are provided in Sect. 4. The conclusion of this paper is reported in Sect. 5.

2 Materials and Methods

This section introduces the materials and methods used in this paper.

2.1 EEG Data Acquisition

The data used in this paper was taken from the publicly available data on the Department of Epileptology at the University of Bonn [12]. This set of data includes five sets (designated A, B, C, D and E), each of which includes 100 single segments EEG of 23.6 s Length, with a sampling rate of 173.6 Hz. Where each segment of data contains N = 4097 data points accumulated at intervals of

1/173.61th of 1 s. Data sets A and B consist of segments extracted from EEG surface recordings that were performed in five healthy volunteers using a unified electrodes positioning scheme. The volunteers relaxed in an open-eyes with a wake state (A) and eyes closed (B), respectively. Data sets C, D and E are detected by epileptic subjects via intracranial electrodes for epileptic ictal and interictal activity.

2.2 Discrete Wavelet Transforms (DWT)

There are signals like EEG having the non-stationary and transient characteristics; in such situation timefrequency methods can be used [5]. We used DWT method to extract the individual EEG sub-bands and reconstruct the information accurately because the wavelet transform has the advantages of time-frequency localization, multi-rate filtering, and scale-space analysis DWT can expose more details from the signal in both time and frequency domain precisely. This makes it become a robust tool in biomedical engineering, particularly in epileptic seizure detection.

In this paper, DWT is utilized to analyze the EEG signals into various frequency bands. The DWT decomposes a specific signal into approximation and detail coefficients at the first level. Then the approximation coefficients are additional decomposed into next level of approximation and detail coefficients [13]. In the first stage of the DWT, an LP and HP filters are used to pass the signal concurrently. At the first level, the outputs from low and high pass filters are indicated to as approximation (A1) and detailed (D1) coefficients. The output signals holding half the frequency bandwidth of the original signal can be down-sampled by two due to Nyquist rule. The same procedure can be duplicated for the first level approximation and the detail coefficients fetch the second level coefficients. Through each step of this decomposition process, the frequency resolution is multiple through filtering and the time resolution is split through down-sampling.

2.3 Grasshopper Optimization Algorithm (GOA)

Grasshopper Optimization Algorithm (GOA) is a meta-heuristic technique [14]. For solving optimized problems, it can be applied and achieves excellent results [15]. In nature, The GOA mimics the behaviour of grasshopper swarms. The mathematical model employed to simulate the swarming behaviour of grasshoppers is presented as follows [15]:

$$X_i = S_i + G_i + A_i \tag{1}$$

where X_i defines the position of the i-th grasshopper, S_i is the social interaction, G_i is the gravity force on the i-th grasshopper, and A_i shows the wind advection.

The S component is calculated as follows:

$$S_i = \sum_{j=1\ j\neq i}^{N} s(d_{ij})\widehat{d_{ij}} \tag{2}$$

Where d_{ij} is the distance between the i-th and the j-th grasshopper, calculated as $d_ij = |X_j - X_i|$, N is the number of grasshoppers, and $\widehat{d_{ij}} = \frac{X_j - X_i}{d_ij}$ is a unit vector from the ith grasshopper to the jth grasshopper. The s function, which defines the social forces, is calculated as follows:

$$s(r) = fe^{\frac{-r}{l}} - e^{-r} \tag{3}$$

Where f indicates the intensity of attraction and l is the attractive length scale.

The G component is calculated as follows:

$$G_i = -g\widehat{e_g} \tag{4}$$

Where g is the gravitational constant and $\widehat{e_g}$ shows a unity vector towards the center of earth.

The A component is calculated as follows:

$$A_i = u\widehat{e_w} \tag{5}$$

Where u is a constant drift and $\widehat{e_w}$ is a unity vector in the direction of wind.

Substituting S, G, and A in Eq. 1, this equation can be expanded as follows:

$$X_i = \sum_{j=1\ j \neq i}^{N} s(|X_j - X_i|)\frac{X_j - X_i}{d_ij} - g\widehat{e_g} + u\widehat{e_w} \tag{6}$$

However, this mathematical model cannot be used directly to solve optimization problems, mainly because the grasshoppers quickly reach the comfort zone and the swarm does not converge to a specified point. A modified version of this equation is presented as follows to solve optimization problems:

$$X_i^d = c\left(\sum_{j=1\ j \neq i}^{N} c\frac{ub_d - lb_d}{2} s(|X_j^d - X_i^d|)\frac{X_j - X_i}{d_ij}\right) + \widehat{T_d} \tag{7}$$

where ub_d is the upper bound in the Dth dimension, lb_d is the lower bound in the Dth dimension, $s(r) = fe^{\frac{-r}{l}} - e^{-r}$, $\widehat{T_d}$ is the value of the Dth dimension in the target (best solution found so far), and c is a decreasing coefficient to shrink the comfort zone, repulsion zone, and attraction zone. Note that S is almost similar to the S component in Eq. 1. However, do not consider gravity (no G component) and assume that the wind direction (A component) is always towards a target $\widehat{T_d}$.

Equation 7 shows that the next position of a grasshopper is defined based on its current position, the position of the target, and the position of all other grasshoppers. Note that the first component of this equation considers the location of the current grasshopper with respect to other grasshoppers.

The coefficient c reduces the comfort zone proportional to the number of iterations and is calculated as follows:

$$c = cmax - l\frac{cmax - cmin}{L} \tag{8}$$

Where $cmax$ is the maximum value, $cmin$ is the minimum value, l indicates the current iteration, and L is the maximum number of iterations.

2.4 Support Vector Machine (SVM)

SVM proposed by Cortes and Vapnik [16]. SVM is a powerful classifier in the field of biomedical science for the detection of abnormalities from biomedical signals. SVM is an efficient classifier to classify two different sets of observations into their relevant class. It is capable of handling high-dimensional and non-linear data excellently. On the basis of the structure of training data sets, it helps to predict the important characteristics of unknown testing data. As in this paper, to evaluate the performance of the proposed technique we are having four test cases with two different sets of class so we preferred this classifier for better accuracy results. SVM mechanism is based upon finding the best hyperplane that separates the data of two different classes of the category. The best hyperplane is the one that maximizes the margin, i.e., the distance from the nearest training points. The structural design of the SVM depends on the following: first, the regularization parameter, C, is used to control the trade-off between the maximization of margin and a number of misclassification. Second, kernel functions of nonlinear SVMs are used for mapping of training data from an input space to a higher dimensional feature space. All kernel functions like linear, polynomial, radial basis function and sigmoid having some free parameters are called hyper-parameters. To date, the kernel generally used in Brain-Computer Interface research was the Gaussian or radial basis function (RBF) kernel with width σ [17].

$$K(x, y) = \exp(-||x - y||^2/2\sigma^2) \tag{9}$$

Where, K(x, y) is termed as the kernel function, which is built upon the dot product of two invariant x and y. Suitable trade-off parameter C and the kernel parameter σ are required to train SVM classifier and usually obtained by the K-fold cross-validation technique. The 10-fold cross validation procedure is suitable for evaluating classification accuracy of a classifier in biomedical signals [18]. In this study, 10-fold scheme has been employed to achieve best performance accuracies.

3 The Proposed Classification GOA-SVM Approach

In this study, we proposed GOA-SVM system for the EEG signal classification. The aim of this approach is to optimize the SVM classifier accuracy by automatically estimating the optimal feature subset and best values of the SVM parameters for the SVM model. The proposed classification approach consists of four main phases. In the first phase, EEG data sets (A, B, C, D, and E) are pre-processed by DWT to decompose into five sub-band signals using four levels decomposition. In the second phase, useful features like entropy, skewness, min, max, median, mean, standard deviation, variance, energy and Relative Wave Energy (RWE) are derived from each sub-band of wavelet coefficients. In the third phase, the relevant features are selected from the extracted features and the parameter values (C, σ) of SVM are dynamically optimized by GOA for EEG signals classification. After that, selected features are applied as an input to SVM-RBF classifier for epilepsy classification task with the obtained optimal parameter values. Finally, the obtained results are evaluated using five different measurements such as classification accuracy, sensitivity, specificity, precession, and F-Measure. The overall process of the proposed method is illustrated in Fig. 1.

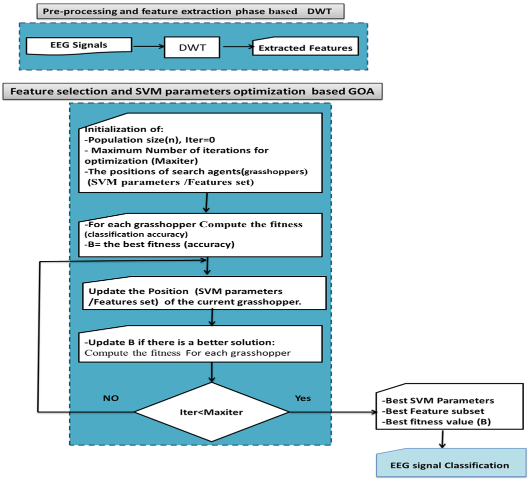

Pre-processing and feature extraction phase based DWT
EEG Signals
DWT
Extracted Features
Feature selection and SVM parameters optimization based GOA
Initialization of:
-Population size(n), Iter=0
- Maximum Number of iterations for optimization (Maxiter)
-The positions of search agents(grasshoppers) (SVM parameters /Features set)
-For each grasshopper Compute the fitness (classification accuracy)
-B= the best fitness (accuracy)
Update the Position (SVM parameters /Features set) of the current grasshopper.
-Update B if there is a better solution:
Compute the fitness For each grasshopper
NO
Iter<Maxiter
Yes
-Best SVM Parameters
-Best Feature subset
-Best fitness value (B)
EEG signal Classification

Fig. 1. Classification approach for SVM based on GOA.

3.1 Pre-processing and Feature Extraction Using DWT

To achieve better results in feature extraction, wavelet decomposition has been used as a pre-processing level for EEG segments to extract five physiological EEG bands, delta (0–4 Hz), theta (4–8 Hz), alpha (8–13 Hz), beta (13–30 Hz), and gamma (30–60 Hz). For this purpose, four levels DWT with fourth order Daubechies (db4) wavelet function have been utilized. Since our dataset is in the range 0–60 Hz, coefficients D1, D2, D3, D4 and A4 corresponding to 30–60 Hz, 15–30 Hz, 8–15 Hz, 4–8 Hz and 0–4 Hz respectively were extracted, that are almost standard physiological sub-bands.

Extracting the features consider the best depict of the behavior of EEG signals and are important for automated seizure detection performance. Feature extraction aims to capture the meaningful and distinctive characteristics hidden in EEG signals, which immediately dominates the final classification accuracy. In our previous work, we have extracted ten features of wavelet coefficients from each sub-band that were chosen to classify EEG signals as shown in [4]. The entire quantitative analysis of EEG signals was coded using MATLAB (R2015a) and the Wavelet function.

3.2 Features Selection and Parameters Optimization Using GOA

Better performance may be achieved by removing irrelevant and redundant data while maintaining the discriminating power of the data by feature selection. The correct selection of relevant features from EEG signals can help the classifier to learn a more robust solution and achieve better generalization performance. In third phase, GOA has the potential to generate both the optimal feature subset and SVM parameters at the same time. SVM tuning parameters have an important impact on their classification accuracy. Improper parameter settings lead to poor classification results. The parameters in SVM that need to be optimized include the penalty parameter C and parameters σ of the radial basis kernel function.

The optimization phase is accomplished in two inner consecutive stages. Each stage either maintains feature set as constant and performs SVM parameters optimization, in this case The best position is the SVM parameters which gives the highest fitness value (average classification accuracy of cross-validation folds in our case), or maintains SVM parameters as constants and performs feature set optimization, in this case, The best position is the subset which gives the highest fitness value.

3.3 Fitness Function

The optimization algorithm generally depends on its fitness function to obtain the best solution. In this paper, the classification accuracy is chosen as the solution qualifier through the search process. Classification accuracy is between the range [0; 1], each grasshopper (Search Agent) reflects a number of accuracies depend on cross-validation strategy. Moreover, each grasshopper reflects ten accuracy values for each fold and all accuracy values for all folds are averaged to return fitness value to the search algorithm as illustrated in the following equation.

$$f(g,t) = \sum_{k=1}^{N} acc_{g,t,k}/N \tag{10}$$

Where $f(g,t)$ the fitness value for grasshopper g in iteration t, N represents the number of folds selected for cross validation and $acc_{g,t,k}$ is the accuracy resultant.

4 Experimental Results and Discussion

4.1 Performance Evaluation Measurements

In this paper, the set A, B, C, and D are considered as positive class and set E is considered as the negative class respectively. To evaluate the classification performance for different test cases in this paper, we have used five measures, which are: (1) Accuracy (Acc), (2) Sensitivity (Sens), (3) Specificity (Spec), (4) Precision (Prec), and (5) F-Measure (F). In general, all mentioned performance measures depend on four main metrics of a binary classification result (positive/Negative); True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN). Where True Positive stands for correctly identified non-seizure activity, True Negative is the correctly identified seizure activity, False Positive is the false identification of non-seizure activity, and False Negative is the falsely recognized seizure activity.

4.2 Experimental Results

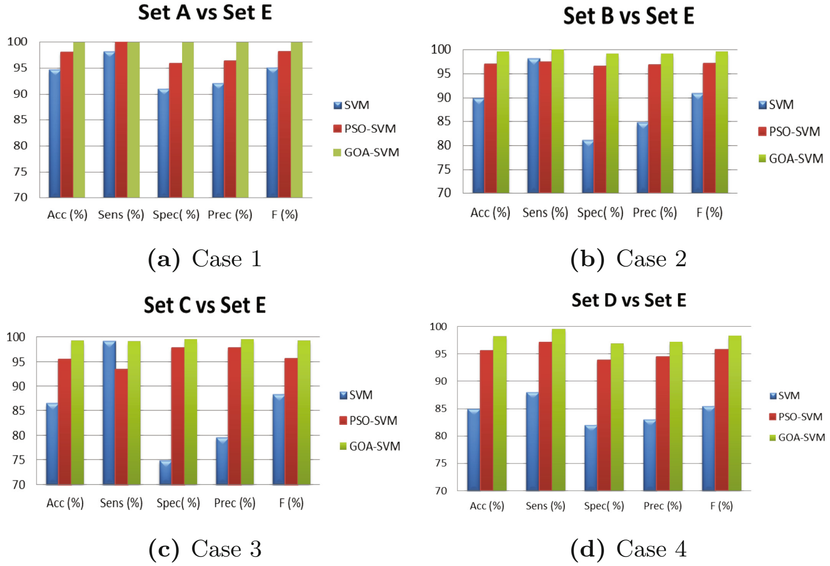
In this paper, the proposed technique is tested on the four different test cases as described in Table 1. The input feature vector is randomly divided into training data set and testing data set based 10-fold cross-validation. The training data set is used to train the classifier, whereas the testing data set is used to verify the accuracy and effectiveness of the trained classifier for the given EEG classification problem. Each row of the input data matrix is one observation and its column is one feature. In this work, the feature vector of data set A has 100 rows and 50 columns. Similarly, the feature vectors of sets B, C, D and E individually have 100 observations and 50 features. The data set for the present binary classifier task consists of 200 observations of 50 features for case 1 to case 4. We implement the GOA-SVM algorithm in MATLAB (R2015a).

The parameter setting for SVM and GOA are No. of agents (No. of grasshoppers) is 30, No. of iteration is 10, cmax is 1, cmin is 0.00004, C in between [1, 1000], σ range is [1, 100] and Feature subset range is [0, 1]. Same number of agents and same number of iterations are used for PSO.

Table 1. The classification description of different test cases along with their EEG data sets.

Test case	Cases for seizure	Classification description
Case 1	Set A vs Set E	Healthy Persons with eye open vs Epileptic patients during seizure activity
Case 2	Set B vs Set E	Healthy Persons with eye close vs Epileptic patients during seizure activity
Case 3	Set C vs Set E	Hippocampal seizure free vs Epileptic patients during seizure activity
Case 4	Set D vs Set E	Epileptic seizure free vs Epileptic patients during seizure activity

The experiments were carried out to assess the performance of the proposed GOA-SVM algorithm for SVM feature selection and SVM parameters Optimization. Classification accuracy rate of the experiment was computed by averaging resultant accuracies from all 10-folds. Figures 2a to d show classification results obtained via applying the proposed GOA-SVM against traditional SVM classification approach and PSO-SVM for RBF kernel function for case 1 to case 4 respectively. As can be seen, the proposed GOA-SVM owns the highest results. Also PSO-SVM is in second place and SVM is the worst one.



(a) Case 1

(b) Case 2

(c) Case 3

(d) Case 4

Fig. 2. SVM, PSO-SVM and GOA-SVM comparative performance measures.

5 Conclusion

In this paper, the discrete wavelet transform is used to analyze EEG to detect epilepsy. EEG signals are decomposed into different sub-bands through DWT to obtain ten features entropy, Skewness, min, max, median, mean, standard deviation, variance, energy and Relative Wave Energy (RWE), from each sub-band to classify EEG signal. This paper develops an approach using GOA for feature selection with SVM parameters optimization and the SVM classifier for automatic seizure detection in EEG signals. The 100% classification accuracies are obtained using GOA-SVM for case 1, 99.577% for case 2, 99.332% for case 3 and 98.268% for case 4. These results illustrate the effectiveness of using GOA and SVM classifier for seizure detection in EEG signals. Experimental results indicated that the proposed GOA-SVMs approach outperformed PSO-SVM and the typical SVMs classification algorithm for RBF kernel function. The proposed method can be used as a quantitative measure to monitor the EEG and may be a useful tool for analyzing the EEG signal associated with epilepsy. We expect this approach would be helpful to both doctors and patients in the course of the illness rehabilitation and diagnosis. As future work, we plan to conduct experiments with more robust classifiers for further investigation in this domain.

References

1. Guo, L., Rivero, D., Dorado, J., Munteanu, C.R., Pazos, A.: Automatic feature extraction using genetic programming: an application to epileptic EEG classification. Expert Syst. Appl. **38**(8), 10425–10436 (2011)
2. Hamad, A., Houssein, E.H., Hassanien, A.E., Fahmy, A.A.: A hybrid EEG signals classification approach based on grey wolf optimizer enhanced SVMs for epileptic detection. In: International Conference on Advanced Intelligent Systems and Informatics, pp. 108–117. Springer (2017)
3. Acharya, U.R., Fujita, H., Sudarshan, V.K., Bhat, S., Koh, J.E.: Application of entropies for automated diagnosis of epilepsy using EEG signals: a review. Knowl. Based Syst. **88**, 85–96 (2015)
4. Hamad, A., Houssein, E.H., Hassanien, A.E., Fahmy, A.A.: Feature extraction of epilepsy EEG using discrete wavelet transform. In: 2016 12th International Computer Engineering Conference (ICENCO), pp. 190–195. IEEE (2016)
5. Kumar, Y., Dewal, M., Anand, R.: Epileptic seizures detection in EEG using DWT-based apen and artificial neural network. Sign. Image Video Process. **8**(7), 1323–1334 (2014)
6. Supriya, S., Siuly, S., Wang, H., Cao, J., Zhang, Y.: Weighted visibility graph with complex network features in the detection of epilepsy. IEEE Access **4**, 6554–6566 (2016)
7. Nicolaou, N., Georgiou, J.: Detection of epileptic electroencephalogram based on permutation entropy and support vector machines. Expert Syst. Appl. **39**(1), 202–209 (2012)
8. Houssein, E.H., Kilany, M., Hassanien, A.E.: ECG signals classification: a review. Int. J. Intell. Eng. Inform. **5**(4), 376–396 (2017)
9. Houssein, E.H., Kilany, M., Hassanien, A.E., Snasel, V.: A two-stage feature extraction approach for ECG signals. In: International Afro-European Conference for Industrial Advancement, pp. 299–310. Springer (2016)
10. Tharwat, A., Hassanien, A.E., Elnaghi, B.E.: A BA-based algorithm for parameter optimization of support vector machine. Pattern Recogn. Lett. **93**, 13–22 (2017)
11. Gaspar, P., Carbonell, J., Oliveira, J.L.: On the parameter optimization of support vector machines for binary classification. J. Integr. Bioinform. (JIB) **9**(3), 33–43 (2012)
12. Department of Epileptology, University of Bonn: EEG time series data. http://www.meb.uni-bonn.de/epileptologie/science/physik/eegdata.html. Accessed Oct 2016
13. Faust, O., Acharya, U.R., Adeli, H., Adeli, A.: Wavelet-based EEG processing for computer-aided seizure detection and epilepsy diagnosis. Seizure **26**, 56–64 (2015)
14. Hassanien, A.E., Emary, E.: Swarm Intelligence: Principles, Advances, and Applications. CRC Press, New York (2016)
15. Saremi, S., Mirjalili, S., Lewis, A.: Grasshopper optimisation algorithm: theory and application. Adv. Eng. Softw. **105**, 30–47 (2017)
16. Cortes, C., Vapnik, V.: Support-vector networks. Mach. Learn. **20**(3), 273–297 (1995)
17. Andrew, A.M.: An introduction to support vector machines and other kernel-based learning methods. Robotica **18**(6), 687–689 (2000)
18. Sharma, R., Pachori, R.B., Gautam, S.: Empirical mode decomposition based classification of focal and non-focal seizure EEG signals. In: 2014 International Conference on Medical Biometrics, pp. 135–140. IEEE (2014)

A Novel Genetic Algorithm Based *k*-means Algorithm for Cluster Analysis

M. A. El-Shorbagy[1](✉), A. Y. Ayoub[1], I. M. El-Desoky[1], and A. A. Mousa[1,2]

[1] Department of Basic Engineering Science, Faculty of Engineering, Menoufia University,
Shebin El-Kom, Egypt
mohammed_shorbagy@yahoo.com

[2] Mathematics and Statistics Department, Faculty of Science, Taif University,
Taif, Kingdom of Saudi Arabia

Abstract. This paper proposed a novel genetic algorithm (GA) based *k*-means algorithm to perform cluster analysis. In the proposed approach, the population of GA is initialized by *k*-means algorithm. Then, the GA operators are applied to generate a new population. In addition, new mutation is proposed depending on the extreme points of clustering. The proposed approach is applied on a set of test problems. The results proved the superiority of the new methodology to perform cluster analysis well.

Keywords: Cluster analysis · *k*-means algorithm · Genetic algorithm

1 Introduction

Cluster analysis is a type of data mining where it divides heterogeneous objects (or cases, observations) into homogeneous groups called clusters. Objects within the same cluster are similar and different from other clusters [1].

Cluster is very important in many fields such as: In market research, companies identify people with similar buying patterns to facilitate their shopping strategies. In the climate, large amounts of weather data have been collected around the world. By dividing these data, there is a clear vision of climatological and environmental trends. In bioinformatics and genetics, cluster analysis is used to identify similar genes, which helps to detect genes responsible for specific genetic diseases [1]. In electrical engineering, Cluster analysis is used to determine the optimal location of the distribution generator [2] and to locate the 3-phase transmission lines fault [3].

There are traditional methods to perform cluster problems which are divided into two methods namely hierarchical method and partitioning method which are described in Sect. 2.

The traditional methods have many disadvantages [4] such as: (1) Empty clusters may be obtained in assignment step as a result of random initial centers. (2) The distribution is not optimal if there are extreme points in the final clusters. (3) There are no rules to determine the number of appropriate clusters.

Due to these limitations, the researchers used the evolutionary methods to solve clustering problem such as genetic algorithm (GA) [5, 6], particle swarm optimization

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 92–101, 2018.
https://doi.org/10.1007/978-3-319-74690-6_10

(PSO) [7], ant colony optimization (ACO) [8], fuzzy optimization [9], simulated annealing (SA) [10], an pigeon-inspired optimization (PIO) [11], monkey algorithm (MA) [12], etc.

GA is a random search method used to obtain the optimal solutions for objective function of an optimization problem depends on the principles of natural selection and genetics [13]. The initial population of GA is randomly selected. Each individual of population of GA called chromosome. The fitness function is evaluated for each individual of population. The solutions are ranking according to the fitness function whether maximum or minimum, and maintain the best solutions. In an extreme GA operators (crossover and mutation) are applied to generate a new population. We provide solutions from the best solutions of the old generation and the new population. The algorithm of GA is terminated when the maximum number of generation has been produced, or the maximum number of the same objective function is achieved [14, 15].

This paper proposes a new methodology to perform cluster analysis based on genetic algorithm (GA). Firstly, the population of GA is initialized by *k*-means algorithm. Secondly, the GA operators are applied. New mutation is proposed depending on the extreme points of clustering to overcome the limitations of *k*-means algorithm.

The paper is organized as follows: Sect. 2 provides a briefly overview of clustering. In Sect. 3 the proposed algorithm is presented. Experimental results are discussed in Sect. 4. Finally, Sect. 5 presents our conclusion.

2 Clustering

The idea of classifying similar objects into groups is a primitive idea that man has known since ancient times. In biology, there is a theory of classification of living organisms known as taxonomy. The classification of animals and plants has played an important role in the fields of biology and zoology [1].

Classification of elements in the periodic table was an important role in understanding the structure of the atom in the 1860s. In astronomy the stars were divided into dwarf stars and giant stars. From here it is clear that classification may involve people, animals, chemical elements, stars, etc., as the entities to be grouped [1].

Cluster analysis is the most important method of classification. Cluster analysis divides objects into small homogeneous groups so that objects within a group are similar in characteristics and different from objects in other groups. Let we have n points represented by the set $\left\{x_1, x_2, \ldots, x_n\right\}$ portioning in k clusters $\left(C_1, C_2, \ldots, C_k\right)$ such that $C_i \neq \phi$ for all $i = 1, 2, \ldots..k$, and $C_i \cap C_j = \phi$ for $i = 1, 2, \ldots..k$, $j = 1, 2, \ldots..k$ and $i \neq j$ [16].

There are traditional methods to perform cluster problems which are divided into two methods namely hierarchical method and partitioning method. Hierarchical clustering has two strategies methods namely agglomerative and divisive. In agglomerative clustering, consider each object to be a separate cluster and then merge each cluster with some until we reach the required clusters. While, divisive clustering, all objects are considered one cluster. By dividing this cluster, one new cluster is created each time. A split operation is repeated until the desired number of clusters is reached [17].

On the other hand, partitioning methods divide to density based clustering and *k*-means method. The widely used *k*-means clustering requires a number of clusters *k* and an initial assignment of data to cluster; the n objects are divided into *k* clusters, such as the minimization of the sum of distance of each element from the center of each cluster [1, 18]. The steps of *k*-means clustering [1] are described as the following:

1. Select initial cluster centers.
2. Each point is assigned to its closest cluster center.
3. Each cluster center is updated to the mean of its elements belonging to the cluster.
4. Repeat steps 2 and 3 till there is no further change in assignment of points to clusters.

3 The Presented Methodology

GA considers one of the evolutionary algorithms that used to obtain the optimal solution of optimization problems based on the principles of natural selection and genetics [19]. GA begins with a set of random initial populations called chromosomes. Then genetics operators (selection, crossover and mutation) are applied to get a new generation. This process is repeated until the maximum number of generation has been produced, or the maximum number of the fitness function is achieved [14, 20]. Figure 1 shows the pseudo code of GA.

```
Initial population is randomly generated.
The fitness function value is calculated for each individual of population.
Do:
    Select parents from the population.
    Genetics operators (Crossover and Mutation) are applied to establish children.
    Calculate fitness for each individual of children.
    Keep the best.
While stopping criterion is achieved.
```

Fig. 1. The pseudo code of GA.

This paper proposes a new methodology to perform cluster analysis based on GA. Firstly, the population of GA is initialized by *k*-means algorithm. Secondly, the GA operators are applied. New mutation is proposed depending on the extreme points in clusters to overcome the limitations of *k*-means algorithm. Each cluster is supposed to be made up of only one element and it is the center, this will never happen and therefore there is error in each cluster. The error in each cluster is defined as the sum of distances between each element and the center of cluster, therefore the fitness function (Sum of squared error) is defined as the sum of the errors of all clusters and Eq. (1) shows how to calculate the value of the objective function mathematically.

$$SSE(C_1, C_2, \ldots, C_k) = \sum_{i=1}^{k} \sum_{x_j \in C_i} \left\| x_j - z_i \right\|; \qquad (1)$$

where $C_1, C_2, \ldots, C_k$ are the clusters to which the data is divided, x_j the data belong to the cluster C_i and z_i cluster center of C_i. The steps of the proposed algorithm are described as follows:

Step 1: Population Initialization
The population is initialized by using *k*-means algorithm to generate only one generation of solution. Each individual is represented by a row-matrix ($1 \times k$) as shown in Fig. 2. Each gene represents the center of each cluster.

Individual	z_1	z_2	z_3	....	z_k

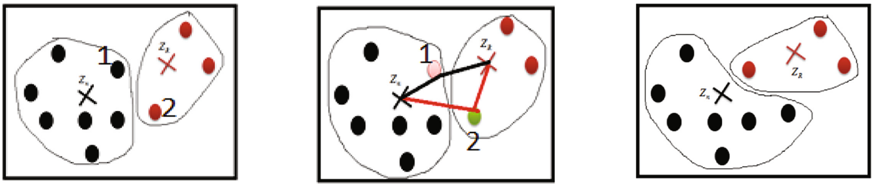
Fig. 2. Structure of the individual.

Step 2: Evaluation
Evaluate the objective function for each individual according to the following Eq. (1)
Step 3: Selection
In the proposed approach, binary tournament selection is applied.
Step 4: Crossover operator
In the proposed approach, single point crossover is employed with probability P_c.
Step 5: Mutation operator
In the proposed approach, we introduced a new mechanism of mutation operator with probability P_m. This mutation depends on the extreme points in clusters. The steps of mutation are described as the following:

- Choose any cluster randomly (C_R)
- Determine the nearest cluster (C_n) to (C_R) according $\|Z_n - z_R\| \prec \|Z_h - z_R\|$; where $h = 1, 2 \ldots\ldots k, h \neq R.$
- Determine the point in (C_R) where its farest from (C_R) and the same time nearest to (C_n) such as $Max. \|X_R - Z_R\|, Min. \|X_R - Z_n\|$
- Determine the point in (C_n) where its farest from (C_n) and the same time nearest to (C_R) such as $Max. \|X_n - Z_n\|, Min. \|X_n - Z_R\|$
- Put $(X_R)\ in\ (C_n),\ (X_n)\ in\ (C_R)$

Figure 3 shows mutation operator in the proposed algorithm. This figure consists of three boxes. The first box shows the clustering before the mutation is performed, while the second box shows the points X_R and X_n and the third shows clustering after the process of mutation.

Step 6: Elitist Strategy
Taking the best individuals from the old population and the new offspring to create the new generation, then go to step 2.
Step 7: Stopping Criterion
The algorithm is terminated when the maximum number of generation has been produced, or the maximum number of the same cluster centers is achieved.



Before mutation

Determination of extreme points

After mutation

Fig. 3. Mutation operator.

4 Experimental Results

To evaluate its performance, the proposed algorithm is used to solve 4 problems containing artificial set (problem 1, problem 2, and problem 3) [5] and problem 4 [21]. In problem 4, there are 4 data sets S1–S4 (two-dimensional artificially generated data sets with varying complexity in terms of spatial data distributions with $k = 15$ predefined clusters). Type of data set, number of points n, number of clusters k and dimension for the four problems are described in Table 1. The proposed algorithm is coded in MATLAB R2013a and the simulations have been executed on an Intel core (TM) i5-3230M CPU @ 2.60 GHZ processor. The parameters adopted in the implementation of the proposed algorithm are listed in Table 2.

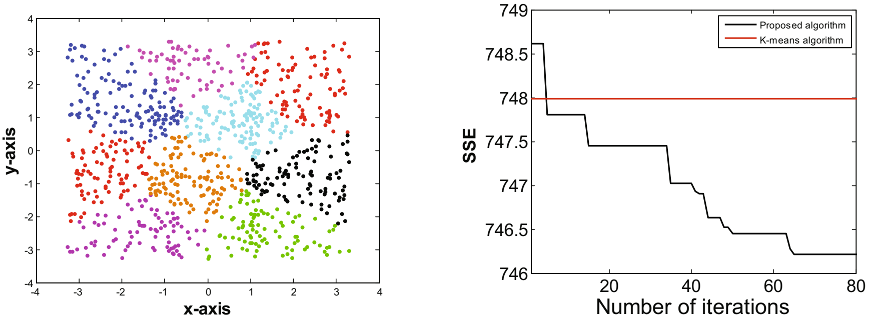
Table 1. Problems contain artificial set.

Problem	Type of data set	n	Number of clusters	Dimension
1	Non over-lapping data	900	9	2
2	Non over-lapping data	300	3	2
2	Non over-lapping data	1000	5	2
3	Artificially generated (S1–S4)	5000	15	2

Table 2. GA parameters.

Crossover rate	0.8
Mutation rate	0.02
Selection operator	Binary tournament
Crossover operator	Single point
Mutation operator	New mechanism
GA generation	10–100

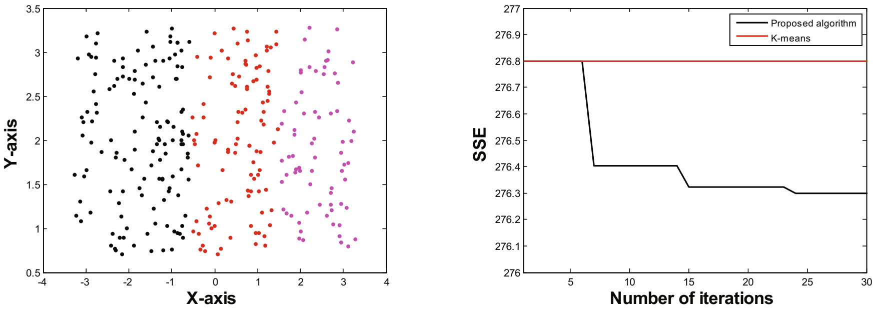
Figures 4, 5, 6, 7, 8, 9 and 10 show the results obtained by the proposed algorithm for all problems. The figures show the distribution of data in clusters and the convergence curve of the fitness function *SSE*. From the figures we can see that the proposed algorithm introduced good distribution of the clusters. In addition, no empty clusters and no extreme points appear in the distribution of our approach. In addition, the proposed approach has fast convergence and gives better results than that obtained by *k*-means algorithm.



Distribution of data

Convergence curve

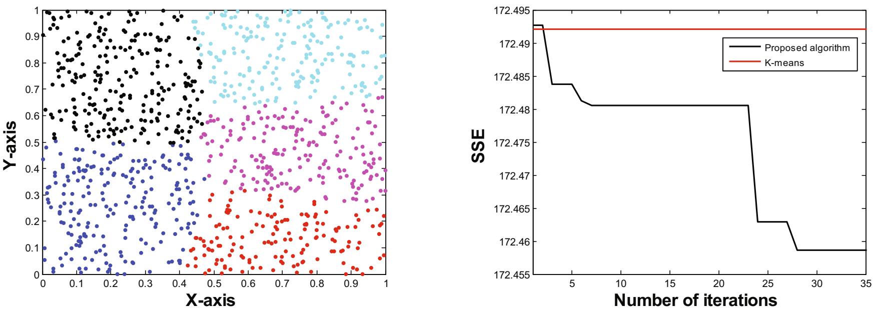
Fig. 4. Results for problem 1.



Distribution of data

Convergence curve

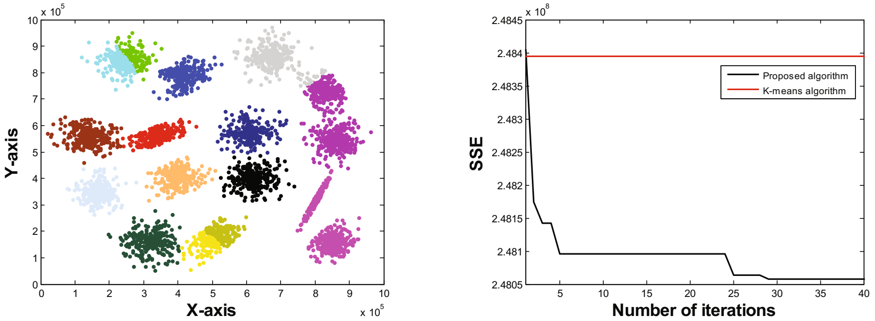
Fig. 5. Results for problem 2.



Distribution of data

Convergence curve

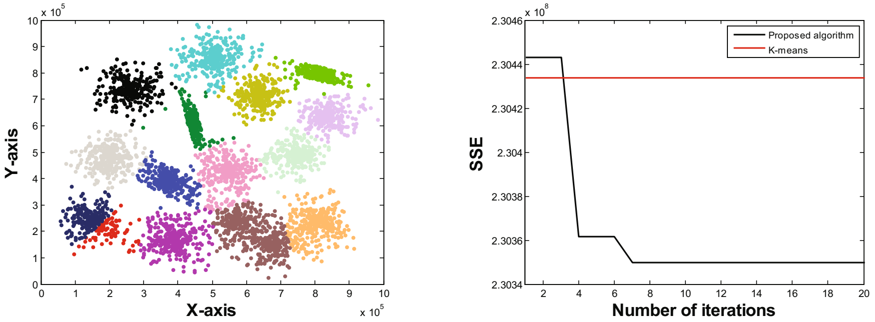
Fig. 6. Results for problem 3.



Distribution of data

Convergence curve

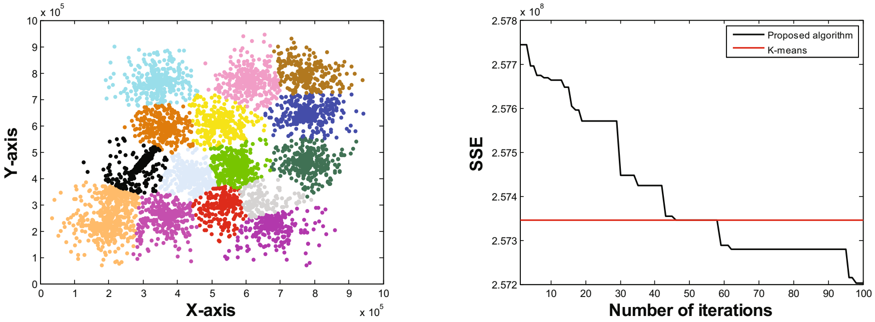
Fig. 7. Results for problem 4:S_1.



Distribution of data

Convergence curve

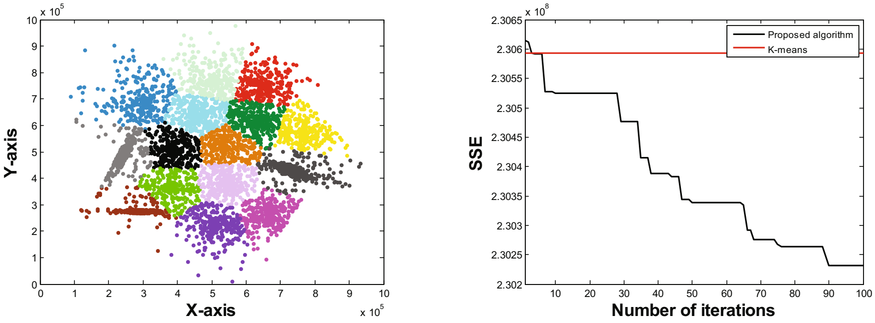
Fig. 8. Results for problem 4:S_2.



Distribution of data

Convergence curve

Fig. 9. Results for problem 4:S_3.



Distribution of data

Convergence curve

Fig. 10. Results for problem 4:S_4.

Table 3. Comparison between *k*-means, and the proposed algorithm with improvement percentage

Problem	*k*		Results of *k*-means	Results of the proposed approach	Improvement percentage
1	9		748.612	746.218	0.32%
2	5		172.492	172.458	0.02%
3	3		232.168	232.131	0.016%
4	15	S_1	248394673.341	248057744.435	0.14%
		S_2	230442886.726	230349740.132	0.04%
		S_3	257743779.908	257203238.855	0.21%
		S_4	230613868.542	230231734.081	0.17%

Table 3 shows the problem, number of clusters *k*, the results obtained by *k*-means, results obtained by the proposed approach and the improvement percentage in the obtained results by the proposed approach in comparison with *k*-means. From the table, we can say that the proposed approach improves the solutions quality in all problems; where the fitness function value obtained by the proposed approach is better than that obtained by *k*-means.

5 Conclusion

In this paper, a new methodology to perform cluster analysis is presented using genetic algorithm (GA). The population of GA is initialized by *k*-means algorithm. Then the GA operators are applied. In addition, a new mutation is proposed depending on the extreme points of clustering. The proposed approach is applied on a set of test problems. The proposed algorithm has many features as the following:

1. It is impossible to have empty clusters as a result of rejecting *k*-means solutions containing empty clusters in the initialization step.
2. There are no extreme points in the final clusters due to the used of the new mutation mechanism.
3. The proposed approach has fast convergence where the number of iterations ranges in all problems from 10 to 100.
4. The value of the fitness function (SSE) obtained by the proposed algorithm is less than that obtained by *k*-means algorithm.

In the future, we aim to apply our approach on a real life application such as image processing.

References

1. Everitt, B.S., Landau, S., Leese, M., Stahl, D.: Cluster Analysis, 5th edn. John Wiley & Sons Ltd., Chichester (2011)
2. Karunakar Jureedi, N.V.V., Rosalina, K.M., Prema Kumar, N.: Clustering analysis and its application in electrical distribution system. Int. J. Electr. Electron. Comput. Syst. **1**, 130–136 (2013)
3. Nithiyananthan, K.: Cluster analysis based fault identification data mining models for 3 phase power systems. Int. J. Innov. Sci. Res. **24**, 285–292 (2016)
4. Singh, K., Malik, D., Sharma, N.: Evolving limitations in K-means algorithm in data mining and their removal. IJCEM Int. J. Comput. Eng. Manag. **12**, 105–109 (2011)
5. Al Malki, A., Rizk, M.M., El-Shorbagy, M.A., Mousa, A.A.: Hybrid genetic algorithm with K-means for clustering problems. Open J. Optim. **5**, 71–83 (2016)
6. Maulik, U., Bandyopadhyay, S.: Genetic algorithm-based clustering technique. Pattern Recogn. **33**, 1455–1465 (2000)
7. El-Tarabily, M., Abdel-Kader, R., Marie, M.: A PSO-based subtractive data clustering algorithm. Int. J. Res. Comput. Sci. **3**, 1–9 (2013)
8. Liu, X., Guangdong, G., Fu, H.: An effective clustering algorithm with ant colony. J. Comput. **5**, 598–605 (2010)

9. Wang, Y., Chen, L.: Multi-view fuzzy clustering with minimax optimization for effective clustering of data from multiple sources. Expert Syst. Appl. **1**, 1–10 (2016)
10. Ranjbar, M., Mosavi, M.R.: Simulated annealing clustering for optimum GPS satellite selection. Int. J. Comput. Sci. **9**, 101–104 (2012)
11. Li, H., Chen, X., Wei, K.: An improved pigeon-inspired optimization for clustering analysis problems. Int. J. Comput. Intell. Appl. **16**, 1–21 (2017)
12. Chen, X., Zhou, Y., Luo, Q.: A hybrid monkey search algorithm for clustering analysis, 1–17 (2014)
13. Filho, J.R., Treleaven, P.C., Alippi, C.: Genetic algorithm programming environments. IEEE Comput. **27**, 28–43 (1994)
14. Alabsi, F., Naoum, R.: Comparison of selection methods and crossover operations using steady state genetic based intrusion detection system. J. Emerg. Trends Comput. Inf. Sci. **3**, 1053–1058 (2012)
15. Mousa, A.A., El-Shorbagy, M.A.: Identifying a satisfactory operation point for fuzzy multiobjective environmental/economic dispatch problem. Am. J. Math. Comput. Model. **1**, 1–14 (2016)
16. Farag, M.A., El-Shorbagy, M.A., El-Desoky, I.M., El-Sawy, A.A., Mousa, A.A.: Genetic algorithm based on k-means-clustering technique for multi-objective resource allocation problems. Br. J. Appl. Sci. Technol. **8**, 80–96 (2015)
17. Franti, P., Kivijarvi, J.: Randomised local search algorithm for the clustering problem. Pattern Anal. Appl. **3**, 358–369 (2000)
18. Mousa, A.A., El-Shorbagy, M.A., Farag, M.A.: K-means-clustering based evolutionary algorithm for multi-objective resource allocation problems. Appl. Math. Inf. Sci. **11**, 1–12 (2017)
19. Farag, M.A., El-Shorbagy, M.A., El-Desoky, I.M., Mousa, A.A., El-Sawy, A.A.: Binary-real coded genetic algorithm based k-Means clustering for unit commitment problem. Appl. Math. **6**, 1873–1890 (2015)
20. El-Desoky, I.M., Nasr, S.N., Hendawy, Z.M., Mousa, A.A., El-Shorbagy, M.A.: A hybrid genetic algorithm for job shop scheduling problems. Int. J. Adv. Eng. Technol. Comput. Sci. **3**, 6–17 (2016)
21. Fränti, P., Virmajoki, O.: Iterative shrinking method for clustering problems. Pattern Recogn. **39**, 761–765 (2006)

Pairwise Global Sequence Alignment Using Sine-Cosine Optimization Algorithm

Mohamed Issa[1,4](✉), Aboul Ella Hassanien[2,4], Ahmed Helmi[1],
Ibrahim Ziedan[1], and Ahmed Alzohairy[3]

[1] Computer and Systems Engineering Department, Faculty of Engineering,
Zagazig University, Zagazig, Egypt
[2] Faculty of Computer and Information, Cairo University, Giza, Egypt
[3] Genetics Department, Faculty of Agriculture, Zagazig University, Zagazig, Egypt
[4] Scientific Research Group in Egypt (SRGE), Cairo, Egypt
http://www.egyptscience.net

Abstract. Pairwise global sequence alignment is a vital process for finding functional and evolutionary similarity between biological sequences. The main usage of it is searching biological databases for finding the origin of unknown sequence. The standard global alignment based on dynamic programming approach which produces the accurate alignment but with extensive execution time. In this paper, Sine-Cosine optimization algorithm was used for accelerating pairwise global alignment with alignment score near one produced by dynamic programming alignment. The reason for using Sine-Cosine optimization is its excellent exploration of the search space. The developed technique was tested on human and mouse protein sequences and its success for finding alignment similarity 75% of that produced by standard technique.

Keywords: Bioinformatics · Sequence alignment · Pairwise global alignment · Meta-heuristics · Sine-Cosine optimization

1 Introduction

Bioinformatics research field merges studies of mathematics, computer science and biology for developing techniques for analyzing biological data [1]. Sequence alignment (SA) is the main process in bioinformatics which is used for measuring the similarity between biological sequences (DNA/protein) to give indications about evolutionary and functional relations between DNA and protein sequences [1, 2]. It compares sequences by matching their bases (nucleotides for DNA and amino acids for protein) to give maximum possible alignment. A lot of biological applications based mainly on SA such as protein secondary structure prediction and analysis [3]. Also, phylogenetic trees construction where the alignment is used to measure the similarity of sequences to produce the evolutionary trees relationship between sequences [4]. DNA fragment assembly uses the sequence alignment to aid in arranging the short, fragmented sequences to construct the original DNA sequence [5].

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 102–111, 2018.
https://doi.org/10.1007/978-3-319-74690-6_11

For aligning two sequences using SA, in this case, is called pairwise alignment while for more than two sequences called multiple sequence alignment [6].

The accelerating sequence alignment algorithm is an important issue due to scanning biological databases for finding the most similar sequences is a time-consuming process. One of the common acceleration techniques is used accelerator hardware devices [7] such as Graphical Processing Units (GPU) [8, 9], Field Programmable Gate Arrays (FPGA) [10, 11] and Multi-Core processing [12, 13]. The main disadvantage of hardware accelerator is its expensive cost.

Another strategy for acceleration is using Stochastic/Meta-Heuristics techniques which aim for finding the best solutions near optimal but in low time than standard approaches [14]. It deals any optimization problem as a black box and by changing the inputs toward the best output found based on the search strategy technique. Meta-Heuristics are inspired its optimization search strategy from the theory of evolutionary, physics and nature. The common meta-heuristics algorithms mimicked from evolutionary are Different Evolution [15], and Genetic Algorithm [16] and from physics are Sine-Cosine Algorithm [17] and Ions Motion Optimization [18]. The swarm-based algorithms which inspired from nature such as Particle Swarm Optimization [19] and Moth-Flame Optimization [20]. Meta-heuristics algorithms succeed in optimizing a lot of applications in different domains such as image processing [21], Parameter tuning of machine learning [22], handwritten recognition [23] and Bioinformatics [24].

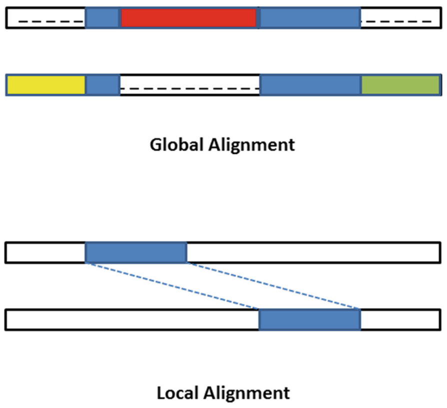
In this paper, pairwise global sequence alignment is optimized using Sine-Cosine optimization algorithm (SCA). SCA is chosen due to its exploration ability of the search space, so the global alignment for long sequences needs high exploration for long sequences. But in contrast, it suffers from the poor exploitation of the search space.

The rest of paper is organized as follow: Sect. 2 describes standard DP pairwise global alignment and details of SCA is detailed in Sect. 3. Section 4 describes the development of global alignment using SCA and the experimental results is detailed in Sect. 5. Finally, conclusion and future work are listed in Sect. 6.

2 Pairwise Global Alignment

2.1 Global Versus Local Alignment

Two kinds of Pairwise alignment are global alignment and local alignment. Figure 1 shows the difference between them where global alignment finds the alignment over the entire lengths of the sequences by aligning the matching regions of the two sequences as in Fig. 1 matching region with blue color and other colors have no matching so aligning it with gaps '-'. To align the entire lengths of sequences gaps must be inserted to shift the matching regions of the two sequences to be aligned. In contrast, Local alignment tries to find the common subsequences between two sequences with high alignment score regardless of the other lengths of sequences as in Fig. 1. The Needle-Wunch global [25] and Smith-Waterman local alignment [26] algorithms based on Dynamic Programming (DP) approach [27] which are the accurate techniques for computing alignment but with extensive execution time especially for sequences with long lengths.



Global Alignment

Local Alignment

Fig. 1. Global alignment vs Local alignment

2.2 Needle-Wunch Global Alignment Algorithm

Global alignment is constructed based on a scoring matrix with size (m + 1) and (n + 1) for row and column respectively where m and n represent the lengths of the two sequences. The score of an alignment from all possible alignments is kept in each cell of the matrix. The alignment score is computed according to (1) based on the previous cells on the same row and column and the diagonal cell.

$$Score(i,j) = \max\left\{\begin{array}{c} Score(i-1,j-1) + Similarity\left(Seq_A(i), Seq_B(j)\right) \\ max_{k=1:i-1}\left(Score(i,k) + g_0 + kg_e\right) \\ max_{k=1:i-1}\left(Score(k,j) + g_0 + kg_e\right) \end{array}\right\} \quad (1)$$

Where *SeqB* and *SeqA* are the sequences to be aligned with lengths *n* and *m* in order, *i* and *j* represent the row and column indices in order, $1 < i < m$ *and* $1 < j < n$. g_o and g_e are the open gap and extended gap penalties where insertion of gaps (*k* is a number of inserted gaps) represents the probability of alignment of bases with nothing for shifting bases for similarity matching. Linear gap penalty ($g_o + kg_e$) is used for scoring sequential gaps instead of separated ones to increase the overall alignment score as possible [28]. *Similarity ()* is used for computing the similarity between two bases according to the used scoring scheme based on un-matching or matching of bases. The common methods for measuring similarity for DNA positive score is assigned for matching nucleotides and un-matching nucleotides negative score is assigned. While for protein sequences two similarity scoring schemes (1) Point Accepted Mutation (PAM) [29] and (2) BLOcked SUbstitution Matrix (BLOSUM) [30]. After computing the scoring matrix, the trace back phase is started to construct the alignment based on scoring matrix computation. Sum of Pair (SOP) scoring function is used to evaluate the overall similarity is used to compute the alignment is as in (2):

$$Alignment_{Score} = \sum_{i=1}^{L} Cost(A(i), B(i)) \tag{2}$$

Where A and B are the aligned sequences and *Cost()* is a function that evaluates the similarity between bases of the aligned sequences. The time complexity of DP local alignment with general gap penalty is $O(m\,n\,\max(m, n))$ and space complexity is $O(mn)$, where m and n are the lengths of the two sequences.

3 Sine-Cosine Optimization Algorithm (SCA)

SCA is a recent population-based optimization technique that based on sine and cosine trigonometric operators for updating the population solutions as in (3).

$$P_i^{t+1} = \left\{ \begin{array}{ll} P_i^t + r_1 \sin\left(r_2\right) \left| r_3 P_{gbest} - P_i^t \right| & r_4 < 0.5 \\ P_i^t + r_1 \cos\left(r_2\right) \left| r_3 P_{gbest} - P_i^t \right| & r_4 \geq 0.5 \end{array} \right\} \tag{3}$$

where r_1 is responsible for balancing between exploration and exploitation and for more exploration, it is assigned high values as in (4).

$$r_1 = a\left(1 - \frac{t}{T}\right) \tag{4}$$

Where T is the maximum number of iterations, t is the current iteration, and a is a constant. r_2 is responsible for the direction of movement towards or outwards P_{gbest} and r_3 controls the effect of destination on current movement. For increasing divergence r_1, r_2, and r_3 are updated at each iteration. While r_4 is used to switch between sine and cosine functions. The steps of SCA are as shown in Algorithm 1.

The time complexity of SCA is $O(T\ N\ C)$ where N is the size of populations and C is the time cost of updating all populations per one iteration.

Algorithm 1. Sine Cosine Algorithm

```
Initialize a set of population solutions ($P_i$), algorithm parameters ($r_1$, $r_2$, $r_3$, and $r_4$)
Repeat
   Evaluate the objective function based on population solution
   Update the best solution obtained so far ($P_{gbest}$)
   Update $r_1$, $r_2$, $r_3$ and $r_4$
  Update the next position of population solutions using (3)
Until (T < maximum number of iterations)
Return the best solution ($P_{gbest}$) obtained as the global optimum
```

4 Global Alignment Using SCA

Each population represents one possible alignment between two sequences where each position of the gap in each sequence with the addition to a number of consecutive gaps is kept. Initially, the difference of lengths of the two sequences is estimated to be gaps inserted in the sequence with short length plus 25% of the shorter sequences is inserted into gaps in each sequence at random position. The gaps are inserted in the form of consecutive and separated ones is chosen randomly to penalize serial gaps to maximize the alignment score. The position of gaps group and the associated number of gaps are kept in variables *P* and *K*. Equation (2) represents the objective function used where the objective is maximizing the alignment score as possible.

The steps for computing global alignment based on SCA as follows:

Step 1: Initialize N possible alignments with random position of gaps in each sequence ($P_A(i)$ and $P_B(i)$) and random number consecutive gaps ($K_A(i)$ and $K_B(i)$) where $1 < i < N$, $1 < P_A(i) <$ length(A) and $1 < P_B(i) <$ length(B). Bound to consecutive number of gaps is determined by the user.

Step 2: Estimate the alignment score for each population *i* according to Eq. (2).

Step 3: Update global best solution (P_{Agbest}, P_{Bgbest}, K_{Agbest} and K_{Bgbest}) of N ones that produce the maximal found alignment score (F_{best}).

Step 4: According to Eqs. (5, 6, 7 and 8) updates the positions of gaps with associated consecutive numbers. Each equation had its parameters (r_1, r_2, r_3) and updated separately of others

$$P_{Ai}^{t+1} = \left\{ \begin{array}{ll} P_{Ai}^{t} + r_1 \sin\left(r_2\right) \left| r_3 P_{Agbest} - P_{Ai}^{t} \right| & r_4 < 0.5 \\ P_{Ai}^{t} + r_1 \cos\left(r_2\right) \left| r_3 P_{Agbest} - P_{Ai}^{t} \right| & r_4 \geq 0.5 \end{array} \right\} \tag{5}$$

$$P_{Bi}^{t+1} = \left\{ \begin{array}{ll} P_{Bi}^{t} + r_1 \sin\left(r_2\right) \left| r_3 P_{Bgbest} - P_{Bi}^{t} \right| & r_4 < 0.5 \\ P_{Bi}^{t} + r_1 \cos\left(r_2\right) \left| r_3 P_{Bgbest} - P_{Bi}^{t} \right| & r_4 \geq 0.5 \end{array} \right\} \tag{6}$$

$$K_{Ai}^{t+1} = \left\{ \begin{array}{ll} K_{Ai_i}^{t} + r_1 \sin\left(r_2\right) \left| r_3 K_{Agbest} - K_{Ai}^{t} \right| & r_4 < 0.5 \\ K_{Ai}^{t} + r_1 \cos\left(r_2\right) \left| r_3 K_{Agbest} - K_{Ai}^{t} \right| & r_4 \geq 0.5 \end{array} \right\} \tag{7}$$

$$K_{Bi}^{t+1} = \left\{ \begin{array}{ll} K_{Bi}^{t} + r_1 \sin\left(r_2\right) \left| r_3 K_{Bgbest} - K_{Bi}^{t} \right| & r_4 < 0.5 \\ K_{Bi}^{t} + r_1 \cos\left(r_2\right) \left| r_3 K_{Bgbest} - K_{Bi}^{t} \right| & r_4 \geq 0.5 \end{array} \right\} \tag{8}$$

Step 5: Repeat from step 2 if T < maximum number of iterations.

The time complexity of SCA global alignment is $O(n\ N\ T)$, where n is the length of the sequences (assume two equal length sequences), N is a number of populations of SCA and T is the number of iterations. This time complexity is lower than cubic one of DP alignment which reflects efficiency of execution time, especially for long sequences.

5 Experimental Results

In each run of SCA alignment, the following measures are calculated on testing the mathematical benchmarking functions:

Statistical mean is the average of solutions that are produced from executing the optimization algorithm for M times and is calculated according to (9).

$$\mathrm{Mean} = \frac{1}{\mathrm{M}} \sum\nolimits_{\mathrm{i}=1}^{\mathrm{M}} \mathrm{S}_{\mathrm{i}} \tag{9}$$

where Si is the optimal solution of the run time i.

Statistical standard deviation (std): is an indicator for the variation of the best fitness values found for running the optimization algorithm for M run times. Also, it represents the robustness and stability, and it is computed as in Eq. (10):

$$\mathrm{Std} = \sqrt{\frac{1}{\mathrm{M}-1} \sum\nolimits_{\mathrm{i}=1}^{\mathrm{M}} \left(\mathrm{S}_{\mathrm{i}} - \mathrm{Mean}\right)^{2}} \tag{10}$$

Table 1. Similarity of protein sequences using DP versus SCA alignment

	Test	Proteins ID	DP score	SCA score			
				Mean	Best	Worst	Std.
Human proteins	T_1	A0PJZ0, Q96IX9	26	16.5	20	14	1.1945
	T_2	Q96IX9, P86434	20	13.9	17	11	1.0546
	T_3	Q96IU4, Q969K4	37	27.0	31	24	1.4434
	T_4	P14060, Q7L8J4	50	39.1	47	34	2.300
	T_5	J3QRE5, H7C0G5	29	21.5	27	19	1.560
	T_6	P04229, P13761	245	156	182	137	9.4348
	T_7	P14060, Q7L8J4	157	85.5	106	60	8.5577
	T_8	E5RI87, A0A1U9X7I0	65	41.8	47	39	1.8592
	T_9	B4DI14, A0A1B0GUD2	30	25.7	29	22	1.3740
Mouse proteins	T_{10}	P05201, Q99JW1	224	112	125	99	6.7504
	T_{11}	Q9Y3Q7, P18089	48	38.5	44	32	2.1113
	T_{12}	O60266, O15204	45	31.9	35	29	1.3887
	T_{13}	Q8R4X1, Q8VD53	32	29.7	34	26	1.6880
	T_{14}	P68510, P63101	148	103	120	82	9.2140

The experimental tests of SCA global alignment were performed on real protein sequences of Homo sapiens (Human), and Mus musculus (Mouse) exists on Swiss-Prot biological databases released October 2017 [31]. The quality of solution was tested by measuring the count of similar bases of two sequences after alignment using Eq. (2) where matching bases increase the count otherwise no effect on the count. Table 1 shows the mean, best, worst and standard deviation of 50 individual runs for protein sequences (Human and Mouse) with different lengths. From Table 1 we can conclude that SCA alignment produce alignment with percent relative to DP alignment score as average 85% for (T9, T11 and T13), 75% for (T2, T3, T4, T5, T12 and T14), 65% for (T1, T6 and T8) and 50% for (T7 and T10). Figure 2 shows the convergence speed of SCA alignment for some tests (T8 – T13). Although the technique has fast convergence, it trapped in local minima that are due to many parameters that need to be tuned and the exploitation of approach is poor.

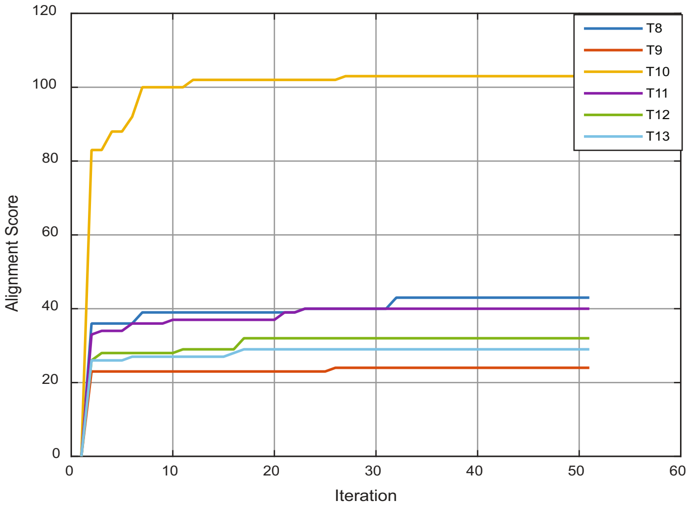


Fig. 2. Convergence of SCA alignment over different tests.

The developed SCA alignment was tested on protein sequences with product lengths (m X n) ranges from 50000 to 7500000. Figure 3 shows the execution time of DP global alignment versus SCA alignment. As shown in Fig. 3 SCA alignment performs faster especially for sequences with longer lengths which reflect the success of the SCA for acceleration of the alignment operation.

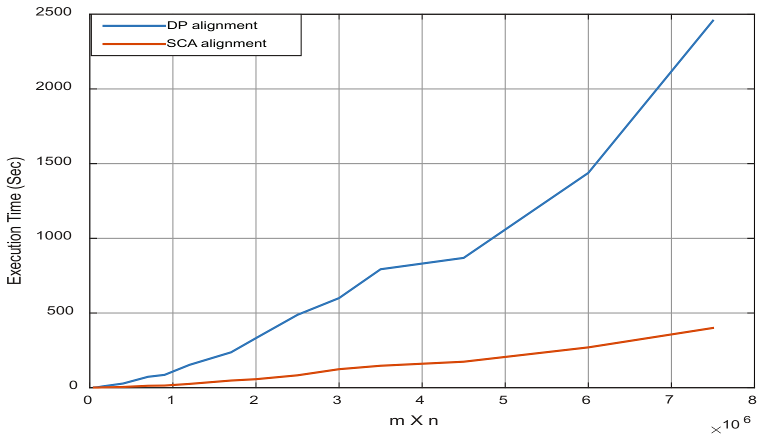


Fig. 3. DP global alignment vs SCA alignment execution time

6 Conclusions and Future Work

In this paper, pairwise global alignment was developed using SCA optimization technique to accelerate its execution for aligning long lengths. The important of global alignment that is used for measuring the functional and evolutionary similarity between two sequences and also scanning databases for finding the origin of unknown sequence. Standard global alignment consumes huge execution time for long sequences. SCA alignment was tested on real protein sequences from Swiss-Prot database released on 2017. The results show that SCA alignment outperforms DP alignment regarding speed but with lower quality of the solution. The SCOA alignment technique is trapping in local minima although its exploration capability. In the future work, SCA alignment is enhanced by merging another meta-heuristic algorithm with exploitation power to balance between exploration and exploitation.

References

1. Cohen, J.: Bioinformatics—an introduction for computer scientists. ACM Comput. Surv. (CSUR) **36**(2), 122–158 (2004)
2. Setubal, J.C., Meidanis, J.: Introduction to Computational Molecular Biology. PWS Pub, Boston (1997)
3. Di Francesco, V., Garnier, J., Munson, P.: Improving protein secondary structure prediction with aligned homologous sequences. Protein Sci. **5**(1), 106–113 (1996)
4. Feng, D.-F., Doolittle, R.F.: [23] Progressive alignment and phylogenetic tree construction of protein sequences. Methods Enzymol. **183**, 375–387 (1990)
5. Li, L., Khuri, S.: A comparison of DNA fragment assembly algorithms. In: METMBS (2004)
6. Xiong, J.: Essential Bioinformatics. Cambridge University Press, Cambridge (2006)
7. Sarkar, S., et al.: Hardware accelerators for biocomputing: A survey. In: 2010 Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS). IEEE (2010)

8. Elloumi, M., Issa, M.A.S., Mokaddem, A.: Accelerating pairwise alignment algorithms by using graphics processor units. In: Biological Knowledge Discovery Handbook: Preprocessing, Mining, and Postprocessing of Biological Data, pp. 969–980 (2013)
9. Issa, M., Helmi, A., Bakr, H.A., Ziedan, I., Alzohairy, A.: Maximizing occupancy of GPU for fast scanning biological database using sequence alignment. J. Appl. Sci. Res. **13**(6), 45–51 (2017)
10. Benkrid, K., Liu, Y., Benkrid, A.: A highly parameterized and efficient FPGA-based skeleton for pairwise biological sequence alignment. IEEE Trans. Very Large Scale Integr. VLSI Syst. **17**(4), 561–570 (2009)
11. Ramdas, T., Egan, G.: A survey of FPGAs for acceleration of high performance computing and their application to computational molecular biology. In: 2005 TENCON 2005 IEEE Region 10. IEEE (2005)
12. Xu, B., et al.: DSA: scalable distributed sequence alignment system using SIMD instructions. In: Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing. IEEE Press (2017)
13. Rognes, T.: Faster Smith-Waterman database searches with inter-sequence SIMD parallelisation. BMC Bioinform. **12**(1), 221 (2011)
14. BoussaïD, I., Lepagnot, J., Siarry, P.: A survey on optimization metaheuristics. Inf. Sci. **237**, 82–117 (2013)
15. Storn, R., Price, K.: Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. **11**(4), 341–359 (1997)
16. Holland, J.H.: Genetic algorithms. Sci. Am. **267**(1), 66–73 (1992)
17. Mirjalili, S.: SCA: a sine cosine algorithm for solving optimization problems. Knowl.-Based Syst. **96**, 120–133 (2016)
18. Javidy, B., Hatamlou, A., Mirjalili, S.: Ions motion algorithm for solving optimization problems. Appl. Soft Comput. **32**, 72–79 (2015)
19. Kennedy, J.: Particle swarm optimization. In: Neural Networks (1995)
20. Mirjalili, S.: Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. Knowl. Based Syst. **89**, 228–249 (2015)
21. El Aziz, M.A., Ewees, A.A., Hassanien, A.E.: Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation. Expert Syst. Appl. **83**, 242–256 (2017)
22. Tharwat, A., Gabel, T., Hassanien, A.E.: Parameter optimization of support vector machine using dragonfly algorithm. In: Hassanien, A.E., Shaalan, K., Gaber, T., Tolba, Mohamed F. (eds.) AISI 2017. AISC, vol. 639, pp. 309–319. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-64861-3_29
23. Abd Elfattah, M., Abuelenin, S., Hassanien, A.E., Pan, J.-S.: Handwritten Arabic Manuscript Image Binarization Using Sine Cosine Optimization Algorithm. In: Pan, J.-S., Lin, J.C.-W., Wang, C.-H., Jiang, X.H. (eds.) ICGEC 2016. AISC, vol. 536, pp. 273–280. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-48490-7_32
24. Ali, A.F., Hassanien, A.-E.: A Survey of Metaheuristics Methods for Bioinformatics Applications. In: Hassanien, A.-E., Grosan, C., Fahmy Tolba, M. (eds.) Applications of Intelligent Optimization in Biology and Medicine. ISRL, vol. 96, pp. 23–46. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-21212-8_2
25. Needleman, S.B., Wunsch, C.D.: A general method applicable to the search for similarities in the amino acid sequence of two proteins. J. Mol. Biol. **48**(3), 443–453 (1970)
26. Smith, T.F., Waterman, M.S.: Identification of common molecular subsequences. J. Mol. Biol. **147**(1), 195–197 (1981)
27. Cormen, T.H.: Introduction to Algorithms. MIT press, Cambridge (2009)

28. Gotoh, O.: An improved algorithm for matching biological sequences. J. Mol. Biol. **162**(3), 705–708 (1982)
29. Henikoff, S., Henikoff, J.G.: Amino acid substitution matrices from protein blocks. Proc. Natl. Acad. Sci. **89**(22), 10915–10919 (1992)
30. Mount, D.W.: Comparison of the PAM and BLOSUM amino acid substitution matrices. Cold Spring Harbor Protocols 2008(6) (2008) https://doi.org/10.1101/pdb.ip59
31. http://www.uniprot.org/ (2017)

An Automated Fish Species Identification System Based on Crow Search Algorithm

Gehad Ismail Sayed[1,3](✉), Aboul Ella Hassanien[1,3], Ahmed Gamal[1,3], and Hassan Aboul Ella[2,3]

[1] Faculty of Computers and Information, Cairo University, Cairo, Egypt
GehadIsmail_FCI@yahoo.com
[2] Faculty of Veterinary Medicine, Cairo University, Cairo, Egypt
[3] Scientific Research Group in Egypt (SRGE), Cairo, Egypt
http://www.egyptscience.net

Abstract. This paper proposed an automated fish species identification system based on a modified crow search optimization algorithm. Median filtering is applied for image smoothing and removing noise through reducing the variation of intensities between the neighbors. Then, a k-mean clustering algorithm is used to segment the fish image into multiple segments. Shape-based and texture-based feature extraction process for classification is presented. A new modified binary version of crow search algorithm is proposed to reduce the data dimensionality of the extracted features. Finally, support vector machine and decision trees are implemented for classification and the fish species are classified based on either their class including Actinopterygii and Chondrichthyes or based on their order. Total of 270 images with different species, classes and orders are used for evaluation of the proposed system. The experimental results show that the proposed system achieves the highest classification accuracy compared to state-of-the-art algorithms. Also, the results show that the overall fish species identification system obtains on average of 10 folds, 96% classification accuracy for classification based on class and 74% for classification based on fish order.

Keywords: Crow Search Algorithm (CSA) · Image classification
Fish identification · Feature selection

1 Introduction

Each family of fishes in the water has physical habits called differentiating characteristics that set it apart from others. These differentiating features help fish outrun in their environment. By monitoring and comparing these features, fisherman learned that fish could affect specific survival and classification into significant collections of identification and another study. Moreover, around the world, there are more than 32 500 species of finfish, the amount of information required to differentiate all of them are too tedious to process; therefore, fish identification is usually conducted at local or regional levels. Increasing globalization of fisheries thus introducing new products to the challenges of aquatic

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 112–123, 2018.
https://doi.org/10.1007/978-3-319-74690-6_12

organisms. Emerging new applications also require precise species identification, and the collection of species for accurate information for sustainable fisheries management has a long traditional nature [19].

Fish classification is the identification of fish species, depending on their characteristics or similarities. Also, it can be defined as the process of determining the types of fish [1]. Classification of fish is necessary for several reasons, including pattern and subsistence matching extraction feature, identification of physical or behavioral characteristics, statistical control and quality applied to fish of all kinds [2]. Moreover, fish classification is considered an important task for fishing and population assessments [3]. However, manually fish classification is very complex and tedious task for those who are not specialists. Fish species are involved in many commercial and agricultural industries, as well as the manufacture of foodstuffs and used as food that is very important to humans [4]. As marine biologists classify fish from their characteristics and also used the classification tree in the classification of fish, which led them to use machine learning and structures in the data Which saved time, effort and speed in the classification of fish [5]. On the other hand, automatic fish classification can accelerate the process and can improve the accuracy of classification or identification of fish species. Several approaches are introduced in literature for automatic fish species identification. Most of these approaches are based on using k-means clustering and support vector Machine (SVM), Fuzzy logic, Artificial Neural Network (ANN) [6].

Authors in [7] proposed an automatic approach for fish identification based on using support vector machine and k-means clustering algorithms. The experimental results show that their identification system obtained 78,59%. Also, they compared their work with the k-nearest neighbor classifier and artificial neural network. Another approach for automatic fish identification is introduced in [8]. In this approach, color and texture based features are extracted from each fish image. Then, these features are used to feed support vector machine. In [9], authors proposed an automatic marine life monitoring system. Their system is composed of three phases. The first phase is to identify fish objects underwater. The second phase is to classify the detected fish species. The third phase is to track the detected fishes and then record the fishes' activities. The authors used support vector machine for fish species identification. They obtained overall 91.7% accuracy. Another automatic Nile Tilapia fish classification approach proposed in [10]. The authors used scale invariant feature transform and robust features for feature extraction. Then, these features are used to feed support vector machine. The experimental results show the efficiency of their approach compared with artificial neural network and k-nearest neighbor.

This paper proposed an automated fish species identification system based on a modified crow search optimization algorithm. Total of 270 images with different species, classes and orders are used for evaluation of the proposed system. The rest of this paper is organized as follows. In Sect. 2, a brief description of the inspiration and mathematical model of CSA is provided. Then, in Sect. 3, the proposed fish species identification model is presented. Section 4 describes the adopted fish species dataset. The experimental results discussed in Sect. 5. Finally, Sect. 6 summarizes the main findings of this paper.

2 Crow Search Algorithm (CSA)

2.1 Inspiration Analysis

The crow search algorithm (CSA) is one of the recently meta-heuristic algorithms proposed in 2016 by Askarzadeh [11]. CSA's inspiration came from the crows mechanism for hiding their food. Crows are clever birds; They have large brain compared with their bodies. Also, they are self-aware birds. They have a good memory to remember faces. Additionally, they used a communication mechanism to warn the other crows, when they found unfriendly animals. Crows sometimes steal the other birds' food. Thus they are called as thieves. They use their experiences in thieving to predict the behavior of pilferer. Crows are known to be very cautious birds, as long they found a thievery, they change the place of hiding their foods [16].

The four main principles of CSA are defined as follows:

- Crows live in the flock's form
- Crows keeps in their memory their hiding places of foods
- Each member of crow follow each others while doing thievery
- Crows are very cautions against thievery, they protect their caches by a probability.

2.2 Mathematical Model of CSA

Suppose M is the number of crows, $y^{j,t}$ is the position of $j-th$ crow at t iteration, where $j = 1, 2, ...M$, D is the number of dimensions, $tMax$ is the maximum number of iterations and $N^{j,t}$ is the hiding place position of hiding place for $j-th$ crow.

At iteration t, $j-th$ crow may follow the hiding place of $z-th$ crow. In this situation, two cases may happen:

Case 1: Crow z doesn't know that crow j follows it: Crow j approaches to the hiding place of crow z. The updating position of crow j is defined as follows:

$$y^{j,t+1} = y^{j,t+1} + R_j \times fl^{j,t} \times (N^{z,t} - y^{j,t}) \tag{1}$$

where R_j is random number $\in [0, 1]$, fl is the flight length.

Case 2: Crow z knows that crow j follows it: Crow z will change its position in the search space to protect its cache.

The previous two cases are mathematically defined as follows:-

$$y^{j,t+1} = \begin{cases} y^{j,t} + R_j \times fl^{j,t} \times (N^{z,t} - y^{j,t}), & R_z \geq AP^{j,t} \\ Choose\ a\ rand\ position, & otherwise \end{cases} \tag{2}$$

where $AP^{j,t}$ is the awareness probability of $z - th$ crow, R_z is random number $\in [0, 1]$.

The original CSA starts with setting the D, AP, M, $tMax$ and fl parameters. Then, it is randomly initialized the positions of crows y. In the beginning, the $y - th$ hide their food at initial positions N as they don't have an experience in hiding their food. Through the optimization process, the position of each crow is evaluated using a specified fitness function. Then, the positions of crows are updated based on the best the fitness value, using Eq. (2). Then, the feasibility of each new crow position is checked and then updates its memory as follows:

$$N^{j,t+1} = \begin{cases} y^{j,t+1}, & Fn(y^{j,t+1})\ is\ better\ than\ Fn(N^{j,t}) \\ N^{j,t}, & otherwise \end{cases} \tag{3}$$

where $Fn()$ is defined as the fitness function.

Finally, the algorithm terminates when the termination criteria is satisfied and the best crow position is reported as the optimal solution.

3 Fish Species Identification System

The proposed fish species identification system is comprised of the following five basic building phases: (1) Pre-processing phase: In the first phase of the investigation, a preprocessing algorithm based on the median filter is presented. It is adopted and used to improve the quality of the images and to make the segmentation and feature extraction phase more reliable, (2) Segmentation phase: In the second phase, a segmentation algorithm using the k-means technique is presented to segment the fish image into multiple segments, (3) Feature extraction phase: Thirty eight shape-based and texture-based features are extracted from the segmented images, (4) Feature selection phase: a new modified binary version of crow search algorithm is proposed to reduce the data dimensionality of the extracted features, and (5) Classification phase: The last phase is the classification and identification of the fish species based on either their class including Actinopterygii and Chondrichthyes or based on their order. These five phases are described in detail in the following section along with the steps involved, and the characteristics feature for each phase and the overall architecture of the introduced approach is described in Fig. 1.

3.1 Pre-processing Phase

At this stage, the medium filtration algorithm is applied to the original fish images to smooth the image and eliminate noise by reducing the difference

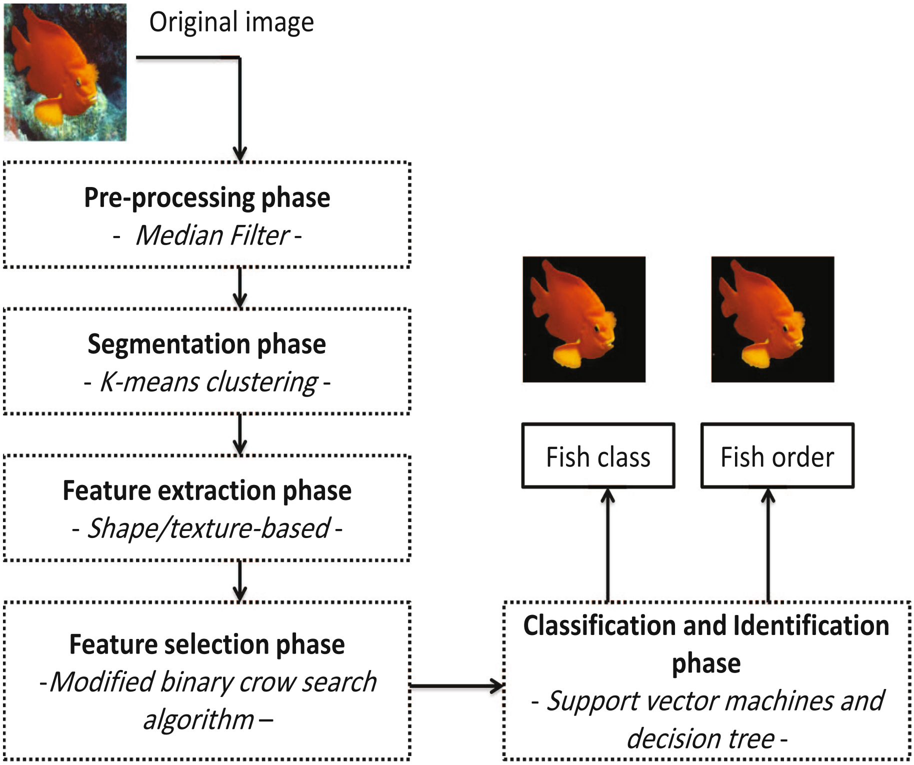


Fig. 1. The proposed fish species identification system architecture

between neighbors' intensity. Image blurring or smoothing is an important task before utilizing the assembly algorithm. In this work, we set the window size of median filter to 13 × 13, where it is selected based on trial and error method. Then, the enhanced fish images are converted to lab color space model. The lab is used to describe all colors visible to the human eye [11]. It has three components; one for lightness L and two for colors a *and* b. The lab is commonly used for digital image manipulation. Also, it is used to remove artifacts and sharp the images.

3.2 Fish Segmentation Phase

The converted image from previous phase is then proceed to the next phase. In this phase, k-mean clustering algorithm is used to segment the fish image into multiple segments. K-means is one of simple and well-known clustering algorithm. The algorithm starts with initializing randomly k points known as centroids, then, based on distance, each data point is assigned to the nearest cluster. This process is repeated until convergence criteria is satisfied [12]. In this work, we set the initial k to 3 (cluster for fish, cluster for background and cluster for other objects in the image). Also, we used Euclidean distance. After partitioning the image into three separate clusters, the best cluster is specified. Later, the best cluster image, which contain the fish objects is selected. Then, the largest connected component of the best cluster image is selected. Thus, all noises in images, which represents small objects resulted from clustering step are removed.

3.3 Feature Extraction Phase

In this phase, two kind of feature categories are extracted from fish object. These two categories are shape-based features and texture-based features. 22 features extracted from grey level concurrence matrix (GRLM). These features are (1) autocorrelation, (2) contrast, (3) correlation, (4) energy, (5) entropy, (6) dissimilarity, (7) inverse difference, (8) homogeneity, (9) cluster shade, (10) maximum probability, (11) sum of squares, (12) sum average, (13) sum entropy, (14) difference variance, (15) difference entropy, (16) cluster prominence, (17), (18) Information measures of correlation-1, (19) Information measures of correlation-2, (20) Maximal correlation coefficient, (21) inverse difference normalized and (22) inverse difference moment normalized. GRLM is one of texture feature methods. It describes the relationship among pixels, which used for estimating the image properties [13]. Also, 16 local and shape features are extracted from segmented image. These features are (1) area, (2) perimeter, (3) eccentricity, (4) equiv Diameter, (5) Euler Number, (6) extent, (7) filled area, (8) major axis length, (9) minor axis length, (10) orientation, (11) solidity, (12) mean, (13) standard deviation, (14) median, (15) skewness and (16) kurtosis [14,16,17].

3.4 Feature Selection Phase

In this phase, after extracting 38 features of the fish segment image, a binary version of crow search algorithm is proposed to select the optimal feature subset. The solutions pool are restricted to $\{0, 1\}$ range in the binary form of crow search algorithm. Equation (4) is used to transfer the agents from the continues form to the binary form [15,16].

$$y^{j,t+1} = \begin{cases} 1 & if(Q(y^{j,t+1})) \geq rand() \\ 0 & otherwise \end{cases} \tag{4}$$

where

$$Q(y^{j,t+1}) = \frac{1}{1 + e^{10(y^{j,t+1} - 0.5)}} \tag{5}$$

where $rand()$ is random number in range $[0, 1]$, generated from uniform distribution and $y^{j,t+1}$ is the updated binary position at t iteration.

In this work, the proposed binary crow search algorithm (BCSA) is employed as a feature selection algorithm based wrapper method. The optimal feature subset is the one, which maximizes the classification performance while minimizes the feature subset length. Next, the detailed description of the proposed algorithm.

Parameters Initialization. The algorithm starts with randomly setting the positions of the crows in the search space and the initial parameters setting. Each position is a feature subset have different length with different number of features. In this work, we set M to 30, AP to 0.1, fl to 2, lb to 0, ub to 1, D to 38 and maximum number of iteration to 70.

Fitness Function. In this step, each crow position is evaluated using a predefined fitness function Fn_t. The best position is the one, which obtained the optimal value. In this work, two evaluation criteria are combined to design the fitness function. These two criteria are feature subset length and classification accuracy. Equation (6) describes the mathematical formula of the adopted fitness function, where Acc denotes the classification accuracy, L_t is the total number of features of the original data (in our case is 38), L_f is the feature subset length and w_f is the weight factor selected in $[0, 1]$. w_F is used to control the importance of an criteria. In this work, we set this parameter to 0.8. Thus, the classification performance is our first priority, then the feature subset length [18]. The original dataset is divided into two datasets parts, namely train and test datasets. Each position is considered as test sample, which will be evaluated. $K - NN$ is the used classifier in the fitness function with Euclidean distance and K equals to 5. Additionally, the fitness function is calculated on average of 3 folds to guarantee the reliability of a crow position (Solution in the search space).

$$Fn_t = maximize(Acc + w_f \times (1 - \frac{L_f}{L_t})) \tag{6}$$

Positions Updating and Termination Criteria. Crows update their positions using Eqs. (1), (2) and (4).

The optimization process is repeated again and again until a predefined criteria satisfied. In our case, we define the termination criteria as when the algorithm reaches the maximum number of iterations.

3.5 Fish Classification Phase

In this section, the optimal feature subset selected from BCSA will be further evaluated. M-fold cross validation method is used, where M set to 10. In this work, two well-know and most common classifiers are used and compared with each other. These classifiers are decision tree (DT) and support vector machine (SVM). The fish species are classified based on either their class including actinopterygii and chondrichthyes or based on their order including atheriniformes, perciformes, cyprinodontiformes, cypriniformes, salmoniformes, carcharhiniformes and tetraodontiformes.

4 Dataset Description

Table 1 shows the description of the adopted fish dataset. In this work, we will focus on classification based on fish class and order. The dataset contains two fish classes, namely actinopterygii and chondrichthyes and 7 fish orders, namely atheriniformes, perciformes, cyprinodontiformes, cypriniformes, salmoniformes, carcharhiniformes and tetraodontiformes.

Table 1. Fish dataset description

Class	Sample Image	No. of fish Orders	No. of Families	No. of Genus	No. of Species
CHONDRICHTHYES		1	4	7	20
ACTINOPTERYGII		7	20	91	240

5 Experimental Results and Discussion

In this section, three main experiments are conducted. The goal of the first experiment is to show the results of preprocessing and clustering phases. The second experiment aims to evaluate the performance of the proposed BCSA, to compare the performance of DT and SVM and to determine the best classifier for the adopted fish dataset. The third experiment aims to compare the performance of BCSA with genetic algorithm (GA) and six other well-known filter and wrapper based feature selection methods. The proposed fish species identification system was tested on 270 fish images with different species. All the experiments performed on the same PC with Core i3 and RAM 2 GB on OS Windows 7. Also, they all programmed in MATLABR 2012.

5.1 Preprocessing and Fish Segmentation Phases Results

Figure 2 shows the obtained results from preprocessing and clustering/fish segmentation phases. Figure 2(a) shows the original fish image, Fig. 2(b) shows the image after applying median filter with window size 13×13. As it can be seen, the image is smoother than the original image. Figure 2(c) shows the enhanced image after converting from RGB to LAB color space model. As it can be observed, the colors of the fish object is brighter or different than the background region. Figure 2(d), (e) and (f) are the clustered images produced from applying k-means clustering algorithms. As it can be observed, the fish object is segmented in a separate cluster.

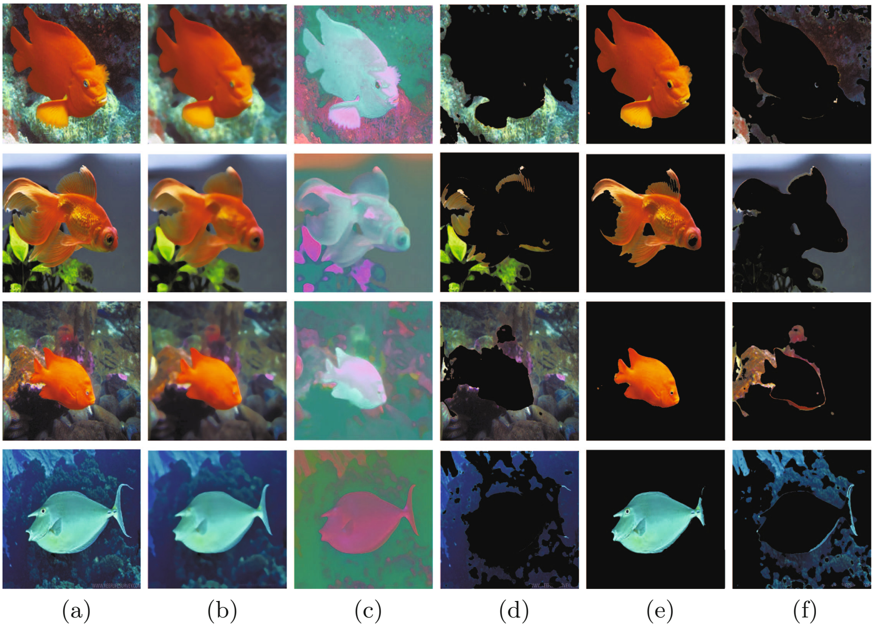


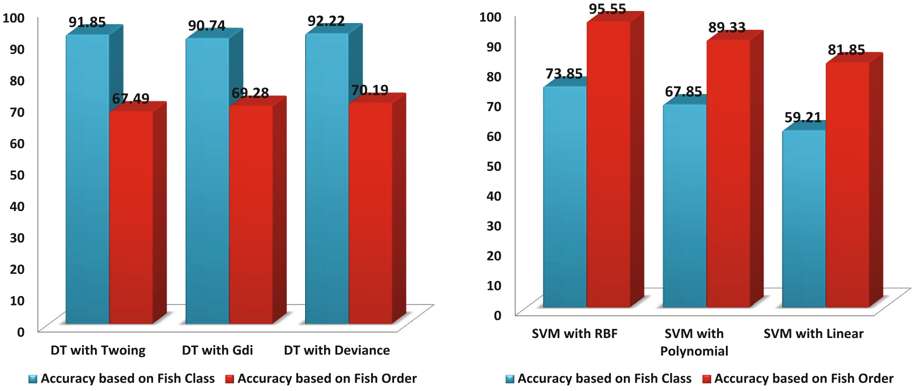
Fig. 2. Preprocessing results; (a) original image, (b) Median filter results, (c) LAB color space results, (d) Cluster-1, (e) Cluster-2 and (f) Cluster-3

5.2 DT vs. SVM Classifier

In this subsection, the performance of decision tree (DT) and support vector machine (SVM) are compared, where different splitting criteria of DT and different kernel functions are considered. The optimal feature subset selected from BCSA is used to feed DT and SVM. Figure 3(a) compares the classification accuracy of the selected feature subset of BCSA using DT with towing, gdi and deviance spiting criteria. As it can be observed, DT with deviance obtains the highest results for both classification categories. Figure 3(b) compares the classification accuracy of selected features from the proposed BCSA using SVM with different kernel functions including RBF, polynomial and linear. As it can be observed, SVM with RBF is the best kernel map. This is due to, SVM with RBF kernel function obtains the highest classification accuracy for both categories.

5.3 BCSA vs. Other Feature Selection Algorithms

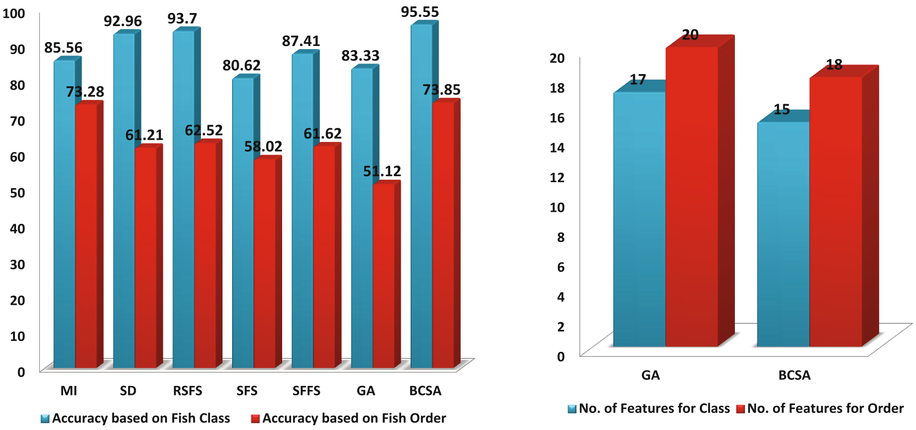
In this section, the performance of the proposed BCSA based wrapper method is compared with different wrapper and filter based feature selection algorithms. In all the following experiments, SVM with RBF kernel function is used in evaluation. Figure 4(a) compares the selected features from the proposed BCSA with other well-known feature selection algorithms in terms of classification accuracy.



(a) (b)

Fig. 3. (a) Comparison between different splitting criteria of decision tree (DT) and (b) Comparison between different kernel functions of support vector machine (SVM)

These algorithms are mutual information (MI), genetic algorithm (GA), random subset feature selection (RSFS), statistical dependency (SD), Sequential Forward Selection (SFS) and sequential floating forward selection (SFFS). As it can be observed, the selected features using BCSA obtains the highest results. Also, RSFS and GA are in second and third place, respectively. Figure 4(b) compares the performance of the proposed BCSA with GA in terms of feature subset length. As it can be observed, the proposed BCSA is superior. It obtains the lowest feature subset length for both classification cases.



(a) (b)

Fig. 4. (a) BCSA vs. different feature selection algorithms and (b) BCSA vs. GA in terms of number of selected features

6 Conclusions and Future Work

This paper introduced a new automated fish species identification system based on crow search algorithm. The proposed fish species identification model is comprised of five main phases; preprocessing phase, fish segmentation phase, feature extraction phase, feature selection phase and fish classification based on either fish class or fish order. Furthermore, in this work, we proposed a binary version of crow search algorithm (BCSA). The proposed BCSA is applied as feature selection based wrapper method. The experimental results show that the proposed BCSA is very competitive feature selection algorithm compared with the other well-known feature selection algorithms. Additionally, the results show the efficiency of the proposed fish species identification system. It obtains overall 96% based fish class and 74% based fish to order. Future work could concentrate on applying the proposed fish species identification system on a larger dataset with more fish species.

References

1. Bridget, B., Junguk, C., Deborah, G., Ryan, K.: Field programmable gate array (FPGA) based fish detection using haar classifiers. In: American Academy of Underwater Sciences, Georgia, USA, pp. 1–8 (2009)
2. Sergio, B.: Fish age classification based on length, weight, sex and otolith morphological features. Fish. Res. **84**(2), 270–274 (2007)
3. Cabreira, A.G., Tripode, M., Madirolas, A.: Artificial neural networks for fish-species identification. ICES J. Mar. Sci. **66**(6), 1119–1129 (2009)
4. Alsmadi, M.K.: Fish recognition based on robust features extraction from size and shape measurements using neural network. J. Comput. Sci. **6**, 1088–1094 (2010)
5. Hoang, T., Lock, K., Mouton, A., Goethals, L.M.: Application of classification trees and support vector machines to model the presence of macroinvertebrates in rivers in Vietnam. Ecol. Inf. **5**(2), 140–146 (2010)
6. Cato, S., Bjorn, T., Darren, W., Overdal, J.: Automatic species recognition, length measurement and weight determination, using the catchmeter computer vision system. In: International Council for Exploration of the Sea, vol. 6, pp. 1–10 (2006)
7. Ogunlana, S., Olabode, O., Oluwadare, S., Iwaskoun, G.: Fish classification using support vector machine. Afr. J. Comput. ICT **8**(2), 1–8 (2006)
8. Hu, J., Li, D., Duan, Q., Han, Y., Chen, G., Si, X.: Fish species classification by color, texture and multi-class support vector machine using computer vision. Comput. Electron. Agric. **88**, 133–140 (2012)
9. Hossain, E., Alam, S.M.S., Ali, A.A., Amin, M.A.: Fish activity tracking and species identification in underwater video. In: 2016 5th International Conference on Informatics, Electronics and Vision (ICIEV), pp. 62–66 (2016)
10. Fouad, M.M., Zawbaa, H.M., El-Bendary, N., Hassanien, A.E.: Automatic Nile Tilapia fish classification approach using machine learning techniques. In: 13th International Conference on Hybrid Intelligent Systems (HIS 2013), pp. 173–178 (2013)
11. Askarzadeh, A.: A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm. Comput. Struct. **169**, 1–12 (2016)

12. Hunter, R.: Photo electric color difference meter. J. Opt. Soc. America **48**(12), 985–995 (1958)
13. Singh, N., Singh, D.: The improved k-means with particle swarm optimization. J. Inf. Eng. Appl. **3**(11), 2224–5782 (2013)
14. Haralick, R., Shanmugam, K., Dinstein, I.: Textural features for image classification. IEEE Trans. Syst. Man Cybern. **3**(6), 612–621 (1973)
15. Agrawal, R.: First and second order statistics features for classification of magnetic resonance brain images. J. Sig. Inf. **3**, 146–153 (2012)
16. Sayed, G.I., Hassanien, A., Azar, A.: Feature selection via a novel chaotic crow search algorithm. Neural Comput. Appl. 1–18 (2017)
17. Sayed, G.I., Hassanien, A.: Moth-flame swarm optimization with neutrosophic sets for automatic mitosis detection in breast cancer histology images. Appl. Intell. **47**(2), 397–408 (2017)
18. Sayed, G.I., Darwish, A., Hassanien, A.: Quantum multiverse optimization algorithm for optimization problems. Neural Comput. Appl. 1–18 (2017)
19. Fish identification tools for biodiversity and fisheries assessments Review and guidance for decision-maker. FAO Fisheries and Aquaculture Technical paper, paper number 545 (2017). http://www.fao.org/docrep/019/i3354e/i3354e.pdf

Fuzzy Logic and Applications

Design and Implementation of Fuzzy PID Controller into Multi Agent Smart Library System Prototype

Hossam Hassan Ammar[1], Ahmad Taher Azar[1,2(✉)],
Tata Derick Tembi[1], Khatthy Tony[1], and Andres Sosa[1]

[1] School of Engineering and Applied Sciences, Nile University, Sheikh Zayed District - Juhayna Square, 6th of October City, Giza 12588, Egypt
{hhassan, T.Derick, T.Khatthy, A.Sosa}@nu.edu.eg,
ahmad_t_azar@ieee.org, ahmad.azar@fci.bu.edu.eg

[2] Faculty of Computers and Information, Benha University, Banha, Egypt

Abstract. This paper compares the performance of four different controllers implemented on two multi agent robots to stabilize its motion from one station to another during delivery tasks. The controllers are; multi-position controller, PID controller, fuzzy logic controller and fuzzy-PID controller. The aim of this paper is to control the mobile robot robustly to arrive its target destination. The robots and station coordinates are recognized using machine vision system and all programming is carried out in LabVIEW. The paper compares the transient response and steady state error of each of controller and experimental results show that the Fuzzy-PID controller produced the best performance and good trajectory of robot from its current position to its target position. It had a better convergence rate when compared with other controllers like PID and Fuzzy logic controllers.

Keywords: Fuzzy logic controller · Fuzzy-PID · Machine vision
Multi agent robot · Multi-position controller · Differential drive

1 Introduction

In the past years, multi agent systems have always been more of a routine in commercial and industrial settings. More attention was being giving to them just because of their capability to move around in real-world environment and accomplish variety of jobs. Recently, multi agent mobile robots' systems are moving out of factories and labs and are becoming more integrated in to everyday life. Although the definition of an agent varies, the common concept is that an agent can be introduced implicitly into the idea that computers can support human beings (Vittorio and Manuela 2015). Multi agent systems could refer to an autonomous proactive and social software component (Iñigo-Blasco et al. 2012; Mizoguchi et al. 2008). Some of these systems are mobile teams of autonomous robots where a set of robots work as a group to attain a common objective. There is still extensive research going on in this fields such as application of artificial intelligence techniques with genetic algorithms, multi robot motion planning, cellular multi agent robot systems implementation to autonomous mobile robots

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 127–137, 2018.
https://doi.org/10.1007/978-3-319-74690-6_13

(Parker 2003). In order for the mobile robot to reach the goal safely, quickly acquired low-resolution and detailed data about the mobile robot's environment is needed for the task execution (Finžgar and Podržaj 2017). For mobile robots to be efficient during their interactions with humans, the implementation of machine vision, computer vision or other image processing algorithms are essential (Mavridis 2015). Vision systems are widely used in industry mainly for inspection and quality control processes, but their use has been increased to robot guidance. Mobile robots need machine vision to move around their working space voiding obstacles and working collaboratively with humans to increase their position accuracy (Pérez et al. 2016).

Achieving intelligent and efficient multi-agent robots requires a robust controller to enable robots carry out their tasks in a robust way especially during its motion. There are so many different opinions about controllers. Some authors consider that artificial intelligence, especially fuzzy control represent a revolution in intelligent control while others believe that what can be achieved using fuzzy methods can also be achieved using conventional techniques, like PID controllers (Suster and Jadlovska 2011). The Conventional PID controller has been widely used in industrial applications particularly due to its ease of use and simplicity of design. The PID controller is limited when dealing with challenging nonlinear systems such as the wheeled mobile robot. External disturbances to the system and changes in system parameters cause undesired performance. To get the desired results, better controllers with flexible and intuitive knowledge based design such as Fuzzy logic and Fuzzy-PID are used. These controllers have proven to be very successful in complex systems were imprecision exists (Erden et al. 2004). The multi agent delivery tasks are sometimes very complex and requires collaboration from other information devices such as sensors, cameras, infrared sensors to provide intelligent support (Soldan et al. 2014). The proposed system consists of two autonomous four-wheel mobile robots moving from one station to the other on a defined work space area. There are four stations with coordinates at the four corner ends of the work space. Each mobile robot has four buttons on it representing a call to a particular station numbered from one through four. The camera is located over the work space to monitor and recognize the station coordinates, robot position, angle orientation of robot, collision avoidance, provide feedback information for the controller and accurately guide the mobile robot. The software interface used for the programming is LabVIEW. The main problem for the mobile robot is to safely move from one station to the other and deliver a package or document in an efficient and safe way. This paper focuses on a reliable way to control the mobile robot motion planning by implementation of a Fuzzy-PID controller on mobile robot. This controller allows the robot to move more smoothly and efficiently from one station to the other. Different controllers such as the multi position controller, PID and Fuzzy Logic controller are also implemented, and the simulation results are verified and compared with the Fuzzy PID controller. The overall organization of this paper is as follows: Sect. 2 provides details about motion planning by differential drive kinematic. Section 3 present the details of the controller design. Section 4 discusses the results with comparative analysis between the different types of controllers implemented. In Sect. 5, concluding remarks and future works are presented.

2 Robot Motion Planning by Differential Drive Kinematic

To navigate and control the motion planning of the robot, there are many methods that can be implemented but the differential drive kinematic is widely used in car-like mobile robots. The configuration of the motion can be represented by $q = (x, y, \theta)$ where x, y is the coordinate of the robot in 2-D plane, whereas θ is the direction angle measuring from the center of the robot to the destination point with respect to the x axis. The equation of motion is represented by (Abatari and Tafti 2013):

$$\dot{x} = v \ \cos\theta \tag{1}$$

$$\dot{y} = v \ \sin\theta \tag{2}$$

$$\dot{\theta} = \omega \tag{3}$$

where v is the linear velocity of the mobile robot, v_r and v_l are the translational velocities of right wheel and left wheel respectively. Assuming that the robot is moving in a curved path with a radius R (Vinod and Mathew 2015), the curved radius of path by left wheel is $R - L/2$ and the curved radius of path by right wheel is $R + L/2$. So,

$$\omega = \frac{v_r}{\left(R + \frac{L}{2}\right)} = \frac{v_l}{\left(R + \frac{L}{2}\right)} = \frac{v_r - v_l}{L} \tag{4}$$

From Eq. (4), the differential drive kinematic is given as

$$\dot{x} = 2(v_r + v_l)\cos\theta \tag{5}$$

$$\dot{y} = 2(v_r + v_l)\sin\theta \tag{6}$$

$$\dot{\theta} = \frac{v_r - v_l}{L} \tag{7}$$

The three equations above can be simplified as:

$$v_r = \frac{v + L\omega}{2R} \tag{8}$$

$$v_r = \frac{v - L\omega}{2R} \tag{9}$$

Where ω is the error between the angle of the robot direction and the angle from robot to the target as shown in Fig. 1, and R is the radius of the robot wheel.

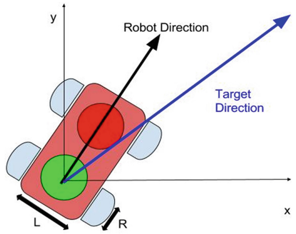


Fig. 1. Robot kinematic diagram.

3 Controller Design

Applying the controller to the system insures the best performance and increases the robustness of the system. Four different controllers are designed to be implemented by starting from the conventional controller which is Multi position and PID. With the simple structure and excellent performance, PID control method is always applied for many applications in closed loop industrial process (Prusty et al. 2014). Then a fuzzy logic (FL) and Fuzzy-PID controllers are designed to visualize and evaluate the performance of the system. The FL is a computational algorithm based on how human's think (Zadeh 1965). The human brain can deal with uncertainties, and judgment and computer can only calculate the precise value. Thus, the FL combine the two techniques together. The Fuzzy-PID or Fuzzy based PID control method is the combination between the conventional PID method and FL method, in which the PID gain parameters are manipulated by the FL algorithm.

3.1 Proposed Multi Position Controller

The output of the system will be changed according to the variation of the error in the interval of value. The output here refers to the speeds of the robot wheel and the changing errors are the result of the difference between the position of the robot and the goal, and the difference between its direction and the goal direction (Rossomando and Soria 2015). It can be presented by:

$$v_r = \begin{cases} 150, & \text{Error distance } > \text{stopping distance} \\ 0, & \text{Error distance } < \text{stopping distance} \end{cases} \tag{10}$$

$$v_l = \begin{cases} 0.8v_r, & \text{Error angle} < -2^o \\ 1.2v_r, & \text{Error angle} > 2^o \\ 0, & -2^o < \text{Error angle} < 2^o \end{cases} \tag{11}$$

From both cases above, v_l mainly depends on v_r which is set manually. The stopping distance is the tolerant value measuring around the goal point.

3.2 Proposed PID Controller

The most famous conventional controller is PID controller (Azar and Serrano 2014, 2015). The controller is configured to improve the guidance of the robot in case of angular velocity error between the direction of the robot and the goal direction while keeping the linear velocity error as constant (Vinod and Mathew 2015). The mathematical model of the control design is given by:

$$c(t) = K_p e(t) + K_p K_i \int e(t)dt + K_p K_d \frac{de(t)}{dt} \tag{12}$$

The input of the controller which is the error signal and the output is the input to the actuator (Prusty et al. 2014). By using machine vision module in LabVIEW, the value of the angle with corresponding to the robot actual direction and the value of the angle with corresponding to the goal can be extracted. The difference between the two angles is the error which is the input of PID controller. From that the angular velocity can be obtained which the unknown parameter in the Kinematic Model. Also, the goal position and the robot position can be extracted which lead to find the linear velocity.

3.3 Proposed Fuzzy Controller

Conventional control design algorithm like PID is based mainly on the mathematical equation to deal with the system behavior. However, it is not easy to use the controller to deal with the nonlinear system. The Fuzzy control design is one of the intelligent control system that based on human think (Zadeh 1965). This kind of control technique can facilitate a complicated system without any knowledge of the system mathematical model (Zadeh 1965). The membership functions of fuzzy input and output parameters has been determined by fuzzy system designer in LabVIEW. The triangle and trapezoidal membership function which are already build-in, have been used in both input and output variables. The system has two input variables. The first variable is the error angle between robot direction and the target direction. The second input variable is the distance error between robot position and the target position. The fuzzy logic controller takes the input variables and compare with the linguistic variables to define the corresponding output to the input variable. The rule of fuzzy logic is formed by a collection of IF-THEN rules which can be also configured in the fuzzy system designer (Prusty et al. 2014).

Both input variables have equally three input variables membership function as shown in Fig. 2. For error angle (EA), the membership functions are described as Turn Left (TL), Turn Right (TR), and Go Straight (GT) as shown in Fig. 3. The determination of these variables are made based on experiment scenario in which the robot need to turn or to go straight. Small (S), Medium (M), Big (B) are the membership function for error distance. The output variables are the velocity of wheel which also have three output variables membership function: Slow, Medium, and Fast. The set of Fuzzy rules are described in Tables 1 and 2 corresponding to the Left Velocity and Right velocity respectively.

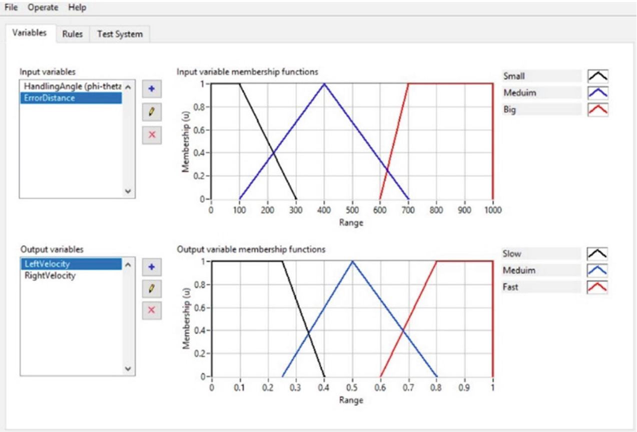


Fig. 2. Fuzzy variables input and output.

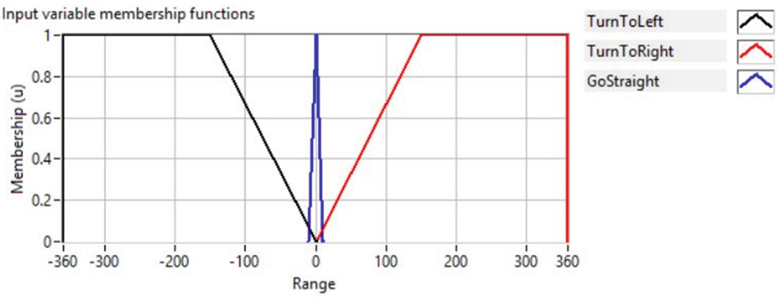


Fig. 3. Membership function of error angle

Table 1. Left velocity fuzzy rules set

Distance error	Angel error		
	TL	TR	GS
S	Slow	Medium	Slow
M	Slow	Fast	Fast
B	Slow	Fast	Fast

Table 2. Right velocity fuzzy rules set

Distance error	Angel error		
	TL	TR	GS
S	Medium	Slow	Slow
M	Fast	Slow	Fast
B	Fast	Slow	Fast

3.4 Proposed Fuzzy-PID Controller

The fuzzy self-adapting PID control is a combination of the fuzzy inference method to automatically adjust the three PID control parameters which is based on the traditional PID control to satisfy system performance and specification (Su et al. 2016). The fuzzy logic is used to tune the PID gain parameter. The parameters of the conventional PID

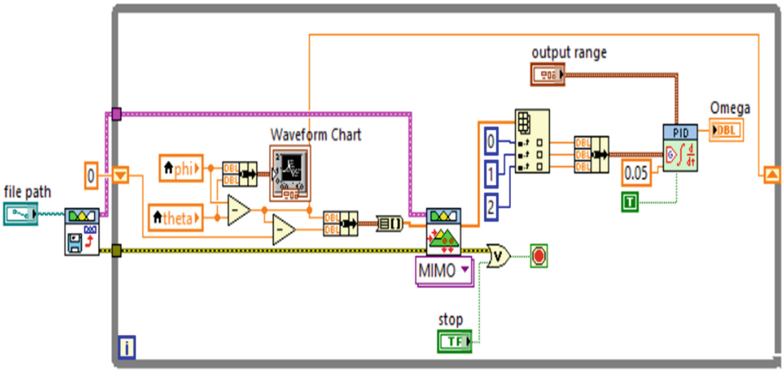
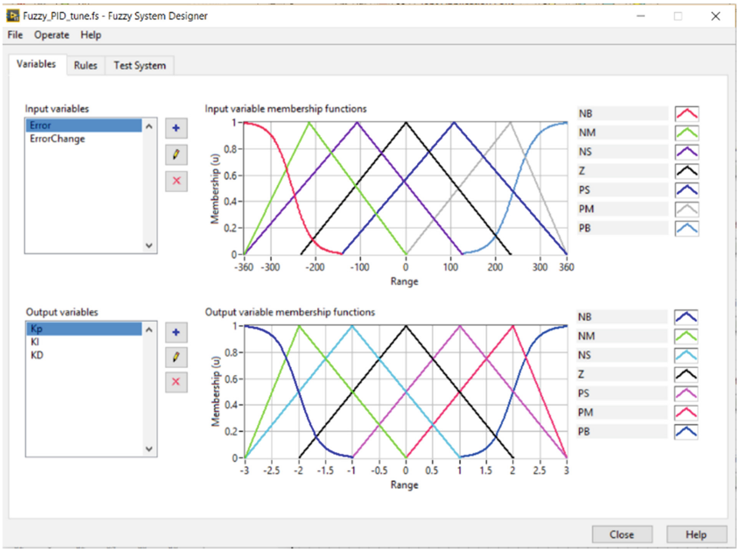


Fig. 4. Fuzzy PID control design in LabVIEW

controller are not often properly tuned for nonlinear parameters variations (Abadi and Khooban 2015; Abatari and Tafti 2013). Using a PID with adjustable parameters might be an efficient solution. Figure 4 demonstrates of fuzzy PID control design in LabVIEW program. The PID gain parameters, which are the values of K_p, K_i, K_d, are considered as output of a fuzzy inference system (FIS) (Abatari and Tafti 2013). The inputs of the Fuzzy logic are angle error (E) and the angle error change (EC) of robot path and target path. The membership function of both inputs is shown in Fig. 5.


Fuzzy_PID_tune.fs - Fuzzy System Designer

Input variables: Error, ErrorChange

Output variables: Kp, KI, KD

Fig. 5. Variable input and output of Fuzzy-PID control design

Table 3. K_p of Fuzzy-PID controller

Error	Change of error						
	NB	NM	NS	Z	PS	PM	PB
NB	NB	NB	NM	NM	NS	Z	Z
NM	NB	NB	NM	NS	NS	Z	Z
NS	NB	NM	NS	NS	Z	PS	PS
Z	NM	NM	NS	Z	PS	PM	PM
PS	NM	NS	Z	PS	PS	PM	PB
PM	Z	Z	PS	PS	PM	PB	PB
PB	Z	Z	PS	PM	PM	PB	PB

Table 4. K_i of Fuzzy-PID controller

Error	Change of error						
	NB	NM	NS	Z	PS	PM	PB
NB	PB	PB	PM	PM	PS	Z	Z
NM	PB	PB	PM	PS	PS	Z	NS
NS	PM	PM	PM	PS	Z	NS	NS
Z	PM	PM	PS	Z	NS	NM	NM
PS	PS	PS	Z	NS	NS	NM	NM
PM	PS	Z	NS	Z	NM	NM	NB
PB	Z	Z	NM	NM	NM	NB	NB

Table 5. K_d of Fuzzy-PID controller

Error	Change of error						
	NB	MM	NS	Z	PS	PM	PB
NB	PS	NS	NB	NB	NB	NM	PS
NM	PS	NS	NB	NM	NM	NS	Z
NS	Z	NS	NM	NM	NS	NS	Z
Z	Z	NS	NS	NS	NS	NS	Z
PS	Z	Z	Z	Z	Z	Z	Z
PM	PB	PS	PS	PS	PS	PS	PB
PB	PB	PM	PM	PM	PS	PS	PB

The input variable membership function range of error is [–360, 360] measured in degree, whereas the error change's input membership function range is [–1, 1]. One of output variable is K_p in which the membership function range is [–3, 3]. Following the same membership function for K_i and K_d which has the range [–0.5, 0.5] and [–1.5, 1.5] respectively. There are seven membership function for both input and output variable, Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z), Positive Small (PS), Positive Medium (PM), and Positive Big (PB) (Prusty et al. 2014). According to the fuzzy sets, the fuzzy rules are described in rule sets table shown from Tables 3, 4 and 5.

4 Result and Discussion

The multi agent robots are placed in random position in the workspace. There are two mobile robots and the robots are given name as R_1 and R_2. Each corner of the workspace defines each destination of the robot to be called. The destination has its own calling button. When the button in one station is pressed, this means that station is calling a robot. The Priority is given to R_2. If R_2 is not available then R_1 takes action. The waiting task option is also created in case the four stations are calling the robot successively. However, the station cannot call a robot twice while it is performing the

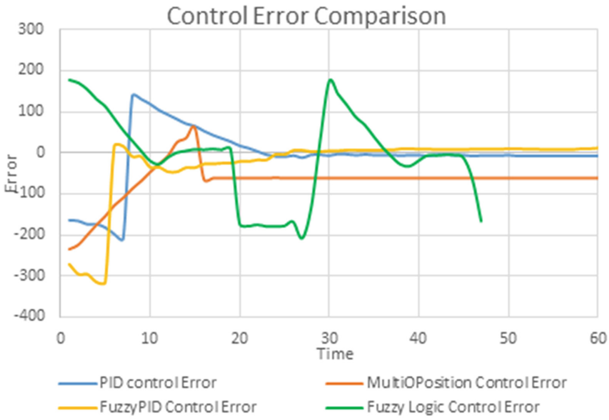


Fig. 6. Combined plot showing control error for different controllers

task of that station until robot is finish the task then the station can call it again. After arriving in the calling station, the robot is ready to take an order which is made by pressing the button on the robot to command it to go to the final destination. When arriving the final destination, the robot will stay and wait for the calling button from the station. The camera is placed on the top of the workspace. It plays a major role as a sensor to capture all the information such as stations coordinates, robots position, robot direction, as well as the destination direction. In this study, the steady state error is mainly concerned as it leads to arrive the destination or the target. Figure 6 depicts all the control design steady state errors. As mentioned, the PID, Fuzzy Logic and Fuzzy PID show the great result as the steady state error is between –10° and 10°.

Table 6 gives a complete detail with respect to specific system parameters when different controllers are implemented.

Table 6. Transient response values for different controllers

Control design	Steady state error (e_{ss})	Overshoot, M_p (%)	Rise time, T_r (s)	Settling time, t_s (s)
PID	−8	7	15	30
Fuzzy logic	8	8.3	9	18
Fuzzy PID	4	4.57	21	30
Multi position	−61	0	1.5	3

Amongst the four controllers, the Fuzzy PID has the lowest steady state error (e_{ss}), the percentage maximum overshoot (M_p) for Fuzzy PID is 3.73% less than that of Fuzzy logic and 2.43% less than the PID controller. The rise time (t_r) is higher than for other controllers but it is tolerable. The Multi position controller has lowest settling time ($_t$s) but its error is so large. Take into consideration that the camera also has a big effect on the result due to the pattern caption and the output caption of the system. The output capture of the machine vision can give only 12 frames per second (fps) while the robots are not busy and only 2 fps when robot is moving. Thus, the speed of the robot is cut down. Moreover, the friction need to be maintained to guarantee the low power consumption.

5 Conclusion

This paper presents the design and implementation of fuzzy-PID controller to control the multi agent robots in a smart library environment prototype. The idea is to put the mobile robot in the service of the library or in the office to serve the user by deliver books or files from table to table or from office to office. Standard controllers are suitable for systems that have an exactly defined mathematical model. In the design of a fuzzy controller it is not necessary to know a precise mathematical model of the plant to be controlled. An important problem of fuzzy controllers is that the computing time is much longer than of a PID, because of the complex operations of fuzzification, inference, and defuzzification. The main purpose of this work is to drive multi agent robots to reach their target or specific goal in response to the different stations button pressed. The rules of Fuzzy PID rules are written to ensure efficient and safe delivery. Three other controllers are used to compare the system response with that of the Fuzzy PID. The simulation results using LabVIEW proved that the Fuzzy PID controller has superior performance including a less overshoot, low steady state error and the robustness. The experiment is conducted using the camera machine vision method to extract some unknown important parameters on the robots as well as the workspace. However, there is one limitation. The frame rate of camera output was really low due to the low specifications of camera. This limitation could be resolved by using a better-quality camera with higher resolution or using multi cameras. Other controllers such as lead lag controllers, PID and other methods, such as neutral network control would be an interesting consideration for future work.

References

Abadi, D.N., Khooban, M.H.: Design of optimal Mamdani-type fuzzy controller for nonholonomic wheeled mobile robots. J. King Saud Univ. – Eng. Sci. **27**(1), 92–100 (2015)

Abatari, H.T., Tafti, A.D.: Using a fuzzy PID controller for the path following of a car-like mobile robot. In: 2013 First RSI/ISM International Conference on Robotics and Mechatronics (ICRoM), Tehran, Iran, 13–15 February 2013

Azar, A.T., Serrano, F.E.: Robust IMC-PID tuning for cascade control systems with gain and phase margin specifications. Neural Comput. Appl. **25**(5), 983–995 (2014). https://doi.org/10.1007/s00521-014-1560-x

Azar, A.T., Serrano, F.E.: Design and modeling of anti wind up PID controllers. In: Zhu, Q., Azar, A. (eds.) Complex System Modelling and Control Through Intelligent Soft Computations. STUD FUZZ, vol. 319, pp. 1–44. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-12883-2_1

Erden, M.S., Leblebicioglu, K., Halici, U.: Multi-agent system-based fuzzy controller design with genetic tuning for a mobile manipulator robot in the hand over task. J. Intell. Rob. Syst. **39**(4), 287–306 (2004)

Finžgar, M., Podržaj, P.: Machine-vision-based human-oriented mobile robots: a review. J. Mech. Eng. **63**(5), 331–348 (2017)

Iñigo-Blasco, P., Diaz-del-Rio, F., Romero-Ternero, M.C., Cagigas-Muñiz, D., Vicente-Diaz, S.: Robotics software frameworks for multi-agent robotic systems development. Robot. Auton. Syst. **60**(6), 803–821 (2012)

Mavridis, N.: A review of verbal and non-verbal human–robot interactive communication. Robot. Auton. Syst. **63**, 22–35 (2015)

Mizoguchi, F., Nishiyama, H., Ohwada, H., Hiraishi, H.: Smart office robot collaboration based on multi-agent programming. Artif. Intell. **114**(1–2), 57–94 (2008)

Parker, L.E.: Current research in multi robot systems. Artif. Life Robot. **7**(1–2), 1–5 (2003)

Pérez, L., Rodríguez, Í., Rodríguez, N., Usamentiaga, R., García, D.F.: Robot guidance using machine vision technique. Sensors **16**(3), 335 (2016)

Prusty, S.B., Pati, U.C., Mahapatra, K.: Implementation of Fuzzy-PID controller to liquid level system using LabVIEW. In: Proceedings of the 2014 International Conference on Control, Instrumentation, Energy and Communication (CIEC), Calcutta, pp. 36–40 (2014)

Rossomando, F.G., Soria, C.M.: Identification and control of nonlinear dynamics of a mobile robot in discrete time using an adaptive technique based on neural PID. Neural Comput. Appl. **26**(5), 1179–1191 (2015)

Soldan, S., Welle, J., Barz, T., Kroll, A., Schulz, D.: Towards autonomous robotic systems for remote gas leak detection and localization in industrial environments. In: Yoshida, K., Tadokoro, S. (eds.) Field and Service Robotics. STAR, vol. 92, pp. 233–247. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-40686-7_16

Suster, P., Jadlovska, A.: Tracking trajectory of the mobile robot Khepera II using approaches of artificial intelligence. Acta Electrotechnica et Informatica **11**(1), 38–43 (2011)

Su, X., Wang, C., Su, W., Ding, Y.: Control of balancing mobile robot on a ball with fuzzy self-adjusting PID. In: 2016 Chinese Control and Decision Conference (CCDC), Yinchuan, pp. 5258–5262 (2016)

Vinod, R.N., Mathew, A.T.: Design, simulation and implementation of cascaded path tracking controller for a differential drive mobile robot. In: 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Kochi, pp. 1085–1090 (2015)

Vittorio, P., Manuela, V.: Handling complex commands as service robot task requests. In: Proceedings of the 24th International Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, pp. 117–118, 25–31 July 2015

Zadeh, L.: Fuzzy sets. Inf. Control **8**(3), 338–353 (1965)

Fuzzy Logic Controller with Color Vision System Tracking for Mobile Manipulator Robot

Ahmad Taher Azar[1,2](✉), Hossam Hassan Ammar[2], and Hazem Mliki[2]

[1] Faculty of Computers and Information, Benha University, Banha, Egypt
ahmad_t_azar@ieee.org

[2] School of Engineering and Applied Sciences, Nile University,
Sheikh Zayed District - Juhayna Square, 6th of October City, Giza 12588, Egypt
hhassan@nu.edu.eg, hazem.mliki@eu4m.eu

Abstract. The purpose of this article is to present a theoretical and practical implementation of a fuzzy algorithm methodology to control a mobile manipulator path planning using a real-time vision system tracking. To meet high performance response and robust stability of the platform navigation, a fuzzy logic controller is designed with realistic constrains. OpenCV library is used to implement Background Modeling technique to track in real time a color object and to extract its (X, Z) coordinates, then an ultrasonic sensor is coupled with the camera to calculate the depth “Y” of the tracked object position. The inverse kinematics is used to control an arm robot to achieve a grasping task of the tracked object. The robot uses the vision system and the ultrasonic sensor to approximate the position of object compared to the cart as well as the position of the arm end effector to the target. The proposed technique shows through simulations and hardware implementation the high efficiency of the algorithm implemented. The fuzzy controller technique presents a good stability and robustness behavior results. The obtained results conclude that the combination between a 2D vision system and an ultrasonic sensor applied to a rigorous fuzzy logic algorithm can perform good results similar to a tracking technique based on a 3D camera.

Keywords: Fuzzy logic controller · Mobile manipulator robot · Machine vision
Color tracking · Background modeling · OpenCV · Inverse kinematics

1 Introduction

The field of mobile manipulator robot application is constantly changing and the greatest challenge tomorrow holds for us to manage this continuity of change [1]. Having prior knowledge on kinematics is very important especially when it comes to animation of articulated structures such as the robot arm [2]. Consequently, a critical step in any robotics system is the analysis and modeling of the kinematics system [3], which is divided into forward and inverse kinematics. The forward kinematics is mainly used to transfer joint variables values to compute the end effector position and the inverse kinematics determines the value of each joint in order to place the arm at a desired position and orientation. The focus of this system is to control a gripper to grasp and manipulate objects in its workspace. In this design, the algorithm is implemented in order to have a flexible framework for the motion planning and the control of the robot.

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 138–146, 2018.
https://doi.org/10.1007/978-3-319-74690-6_14

Mobile robot platforms have become an important part in different aspects of today's modern applications in research and industry [4]. With the advances in controls theory, application of mobile robots in industry has shown growing interest towards automation [5]. The manipulator robot and the mobile cart are combined to navigate and accomplish grasping task in a wide range compering to the limited workspace of the arm robot. To reach the goal, a fuzzy logic algorithm that consider the maximum of constrains in real scenarios is designed and implemented. The design of the controller follows the method of fuzzification, engine inference and defuzzification technic. Rules are refined to meet with realistic scenarios for the mobile manipulator navigation.

A vision system is added to locate the target object inside the space in which the robot can operate. A simple 2D vision system is coupled with an ultrasonic sensor to compute in real time the actual goal objet. First the 2D camera extract the (X, Y) coordinates using background modeling technique, the robot will navigate to the indicated position and then it joins the (X, Z, Y) to approach the final location using the depth extracted from the ultrasonic sensor.

The design challenges of a control system in this regard are the response overshoot, shorter settling time and smaller steady state error.

This article presents a fuzzy controller algorithm. The method discussed in this article try to generate best outputs performances. The controller proposed results with a fast time response and high stability. The obtained results are very promising. The rest of the paper is organized as follows: The mathematical model is presented in Sect. 2. While in Sect. 3, the controller design of the fuzzy logic algorithm is introduced and explained. In Sect. 4, all results are shown under different tests and inputs. In addition, the discussion section and a conclusion is written to conclude this article.

2 Mathematical Model

2.1 Kinematic Model of a Nonholonomic Mobile Robot

The main feature of the kinematic model of wheeled mobile robots is the presence of nonholonomic constraints due to the rolling without slipping condition between the wheels and the ground [6].

The system-generalized velocities w cannot assume independent values; in particular [7], they must satisfy the constraint entailing that the linear velocity of the wheel center lies in the body plane of the wheel, which is the zero lateral velocity [8] (Fig. 1):

$$\mathrm{P} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}, \; U = \begin{bmatrix} v \\ w \end{bmatrix} \tag{1}$$

$$ICC = (x - Rsin(\theta), \mathrm{y} + Rsin(\theta)) \tag{2}$$

$$w\left(R + \frac{L}{2}\right) = V_R \quad \textbf{and} \quad w\left(R - \frac{L}{2}\right) = V_L \tag{3}$$

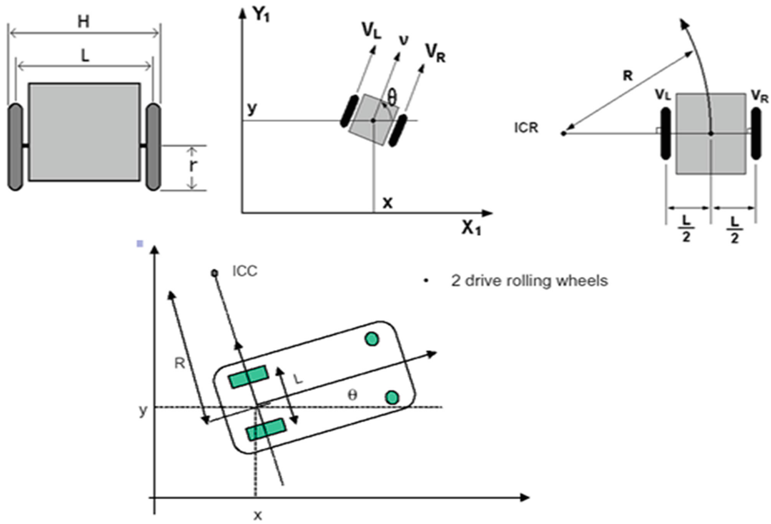


Fig. 1. Kinematic model of a nonholonomic mobile robot

The relation between the control input and the speed of wheels these equations is determined:

$$V_R = rw_R \quad \textbf{and} \quad V_L = rw_L \tag{4}$$

$$w = \frac{V_R - V_L}{L} \quad \textbf{and} \quad v = \frac{V_R + V_L}{2} \tag{5}$$

$$\begin{bmatrix} \mathrm{x} \\ \dot{\mathrm{x}} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \tag{6}$$

The system is subject to two nonholonomic constraints, one for each wheel.

$$\begin{bmatrix} \sin(\theta) & -\cos(\theta) & 0 \end{bmatrix} \begin{bmatrix} \mathrm{x} \\ \dot{\mathrm{x}} \\ \dot{\theta} \end{bmatrix} = 0 \tag{7}$$

$$\dot{x} \sin(\theta) - \dot{y} \cos(\theta) = 0 \tag{8}$$

$$\begin{bmatrix} V_x(t) \\ V_y(t) \\ \dot{\theta}(t) \end{bmatrix} = \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ 0 & 0 \\ \frac{-r}{L} & \frac{-r}{L} \end{bmatrix} \begin{bmatrix} w_l(t) \\ w_r(t) \end{bmatrix} \tag{9}$$

$$R = \frac{L}{2}\frac{V_R + V_{Rl}}{V_R - V_L} \tag{10}$$

2.2 Mathematical Modeling of a DC Motor

A common actuator in control systems is the DC motor [9]. It directly provides rotary motion and, coupled with wheels or drums and cables, can provide translational motion. The electric equivalent circuit of the armature and the free-body diagram of the rotor are shown in the following Fig. 2 [10].

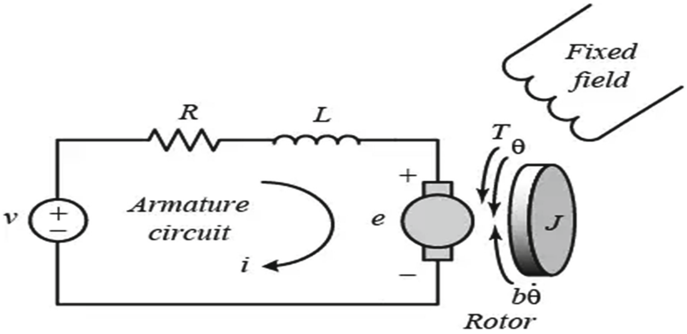


Fig. 2. Model representation of a DC motor

The torque generated by a DC motor is proportional to the armature current [11] and the strength of the magnetic field. In this case, the magnetic field is assumed to be constant [12] and, therefore, that the motor torque is proportional to only the armature current i by a constant factor K_t [13] as shown in the equation below (Table 2).

$$T = K_t i \tag{11}$$

$$e = K_e\dot{\theta} \tag{12}$$

In SI units, Kt = Ke; therefore, K is used to represent both the motor torque constant and the electromotive force.

$$J\ddot{\theta} + b\dot{\theta} = ki \tag{13}$$

$$L\frac{di}{dt} + Ri = V - K\dot{\theta} \tag{14}$$

From Eqs. (13) and (14), the Laplace transform is applied and the results are shown by the modeling equations:

$$s(Js + b)\Omega(s) = kI(s) \tag{15}$$

$$s(Ls + R)I(s) = V(s) - Ks\Omega(s) \tag{16}$$

$$\frac{\Omega(s)}{V(s)} = \frac{K}{(Js + b)(Ls + R) + K^2} \tag{17}$$

3 Controller Design

The design target of the system is to control the cart robot with a fuzzy logic algorithm in same time with the arm robot [14]. Therefore, the design task is divided into two parts. The design process is started by the cart robot controller to perform good navigation results.

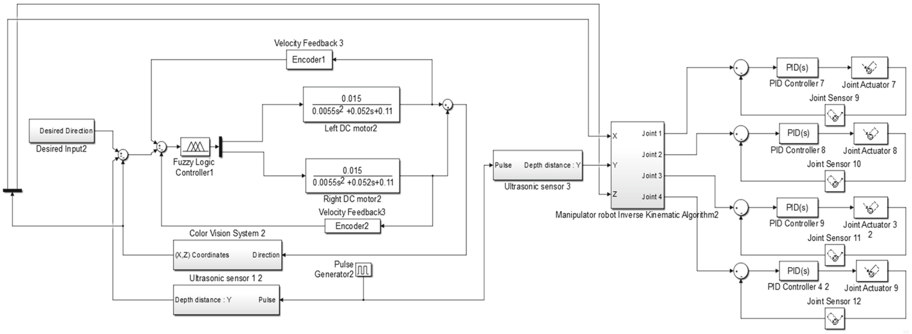


Fig. 3. Block diagram of mobile manipulator robot based Fuzzy logic controller

Table 1. Physical parameters of nonhlonomic robot

x	Cartesian coordinate of the front wheel
y	Cartesian coordinate of the front wheel
V	Linear velocity
θ	Orientation of the robot
w	Angular velocity
L	The distance between the wheels
r	Radius of each wheel
R	Instantaneous curvature radius of the robot trajectory
ICC	Instantaneous center of curvature
v_R, v_L	He linear velocity of the right wheel and left wheel respectively

A typical structure of a fuzzy logic controller is shown in Fig. 4. Using a preprocessor, the inputs that were in the form of crisp values generated from feedback error (e) and change of error (de) [15] were conditioned in terms of multiplying by constant gains before entering into the main control block. The fuzzification block converts input data to degrees of membership functions and matches data with conditions of rules. From the rule based commands, the Mamdani-type inference engine determined the capability of degree of employed rules and returned a fuzzy set for defuzzification block where the fuzzy output data were taken and crisp values were returned (Fig. 3).

The outputs of the fuzzy sets were converted to crisp values through centroid fuzzification method [16]. The post-processing block then converted these crisp values into standard control signals [17]. In this project, experiential knowledge was borrowed from proportional integral control error and change of error to define fuzzy membership functions. The rule Table 1 was then designed and used with a triangular membership function inputs-output in the fuzzy logic controller and was implemented in the simulation (Figs. 4 and 5).

Table 2. Physical parameters of DC motor

J	Moment of inertia of the rotor (kg.m^2)
b	Motor viscous friction constant (N.m.s)
K_t	Electromotive force constant (V/rad/sec)
K_e	Motor torque constant (N.m/Amp)
R	Electric resistance (Ohm)
L	Electric inductance (H)
$\dot{\theta}$	Angular velocity of the shaft

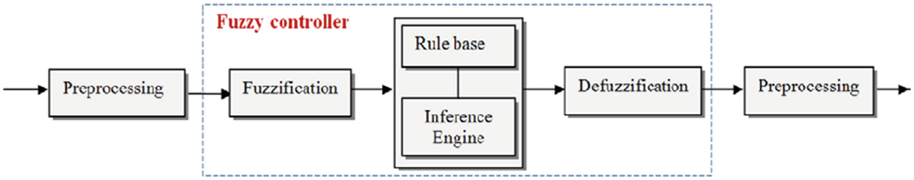


Fig. 4. Fuzzy logic controller block diagram

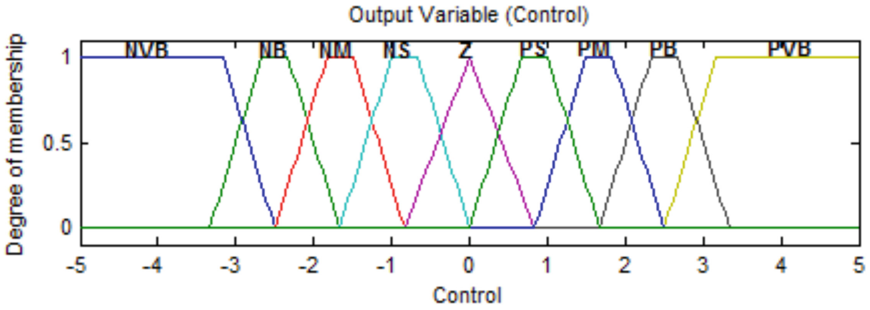


Fig. 5. Example of Fuzzy logic output variable, control

These rules make control efforts based on several if-then statements about (e) and (de), i.e., if the error is equal Negative Big (NB) and change of error is equal to negative medium (NM), then the change in control (c) is positive big (PB). The numbers of these if-then statements were determined based on experiment and tuning of the system. Plots of fuzzy logic membership function *f* the output (c) is shown in Fig. 6 (Table 3).

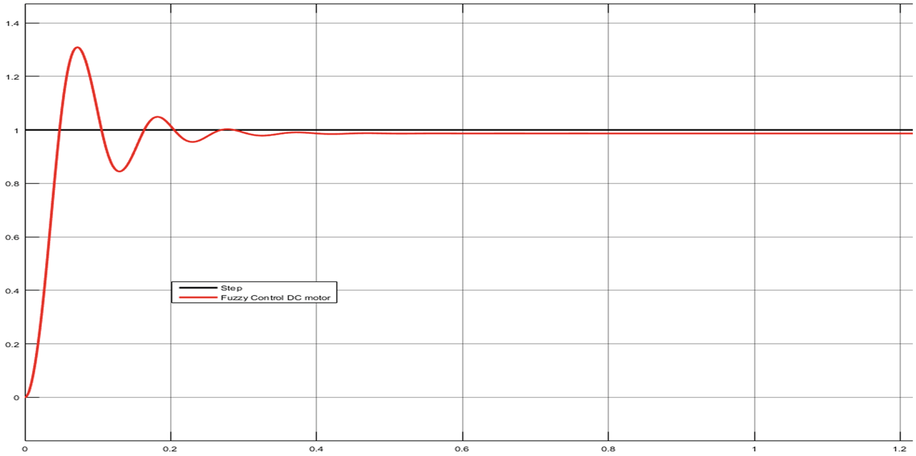


Fig. 6. Step response of the Fuzzy logic controller

Table 3. Fuzzy logic controller rules table

De/e	NVB	NB	NM	NS	Z	PS	PM	PB	PVB
NVB	PVB	PVB	PVB	PB	PM	PM	PS	Z	Z
NB	PVB	PVB	PB	PM	PS	PS	PS	Z	Z
NM	PVB	PB	PM	PS	PS	Z	Z	Z	NS
NS	PB	PM	PM	PS	PS	Z	Z	NS	NS
Z	PM	PM	PS	Z	Z	Z	NS	NS	NM
PS	PM	PS	PS	Z	NS	NS	NM	NM	NB
PM	PS	PS	Z	NS	NS	NM	NB	NB	NB
PB	PS	Z	Z	NS	NM	NM	NB	NVB	NVB
PVB	Z	Z	NS	NM	NM	NB	NB	NVB	NVB

4 Results and Discussion

4.1 Cart Robot Controller's Results

The performances of the fuzzy logic controller are simulated in MATLAB© and also implemented in real mobile manipulator robot. A signal generator produces input references for each control blocks. The fuzzy logic controller block processes the inputs, output of fuzzy inference engine, and generate control signal to control the DC motor dynamic model. The behavior of the closed loop response and the performance of the controllers were evaluated by input step functions with results plotted in Figs. 6 and 7.

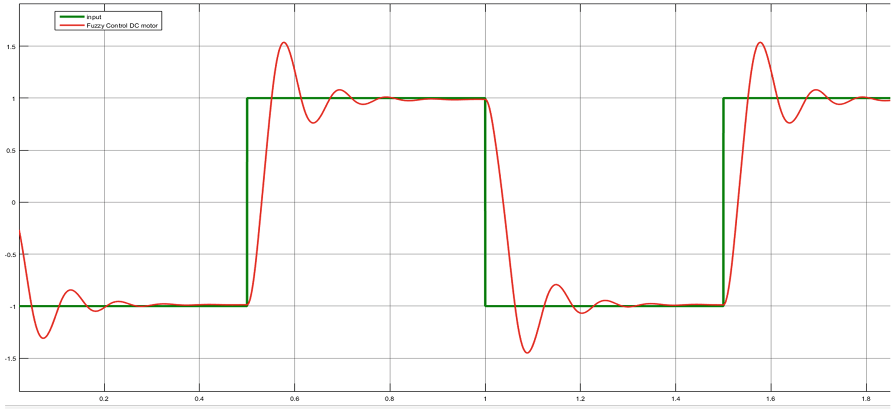


Fig. 7. Fast change input response of the Fuzzy controller

The above simulations show that the fuzzy controller can satisfactorily control a variety of processes. It yields a good control performance, which is confirmed by comparing performances indexes such as the percent maximum overshoot, settling time, and the stability. The Fuzzy logic controller is quietly faster however, it has a problem of maximum overshoot, which exceeds 27%. In addition, it has a permanent steady state error as shown in Fig. 6.

5 Conclusion

The mobile manipulator platform requires precise autonomous devices to perform labor-intensive task such as data collections and image acquisition. This study discussed about simulation and analysis of the fuzzy design for speed control of a DC motor actuator that was used in a mobile robot platform, which moves between crop rows to collect image data and to track an object due to its color. A linear differential equation describing the electromechanical properties of a DC motor to model the relation between voltage input and shaft rotation output was first developed using basic laws of physic. This transfer function was then used to analyze the performance of the system and to design proper controllers to meet the design criteria. To achieve smoother control, a fuzzy logic controller with two inputs and one output including was designed. The results showed that for rectangular changes of the robot speed, the fuzzy logic controller has a good performance in terms of rise time.

References

1. Albu-Schäffer, A., Haddadin, S., Ott, C., Stemmer, A., Wimböck, T., Hirzinger, G.: The DLR lightweight robot: design and control concepts for robots in human environments. Ind. Robot **34**(5), 376–385 (2007)
2. Bøgh, S., Hvilshøj, M., Myrhøj, C., Stepping, J.: Future Manufacturing Assistant – The Mobile Robot “Little Helper”, Master thesis, Aalborg University (2008)

3. EUROP: Robotics visions to 2020 and beyond, The Strategic Research Agenda for Robotics in Europe (2009)
4. Hamner, B., Koterba, S., Shi, J., Simmons, R., Singh, S.: An autonomous mobile manipulator for assembly tasks. Auton. Robot **28**, 131–149 (2009)
5. Hentout, A., Bouzouia, B., Akli, I., Toumi, R.: Mobile Manipulation: A Case Study, Robot Manipulators, New Achievements (2010). ISBN 978-953-307-090-2
6. Ma, Y., Liu, Y., Wang, C.: Design of parameters self-tuning fuzzy PID control for DC motor. In: Second International Conference on Industrial Mechatronics and Automation (ICIMA), vol. 2, pp. 345–348, 30–31 May 2010
7. Naik, K.A., Shrikant, P.: Stability enhancement of DC motor using IMC tuned PID controller. Int. J. Adv. Eng. Sci. Technol. **4**(1), 092–096 (2011)
8. Haung, G., Lee, S.: PC based PID speed control of DC motor. In: 2008 International Conference on Audio, Language and Image Processing, Shanghai, pp. 400–407 (2008). https://doi.org/10.1109/icalip.2008.4590052
9. Anatolii, S., Naung, Y., Oo, H.L., Khaing, Z.M., Ye, K.Z.: The comparative analysis of modelling of simscape physical plant system design and armature-controlled system design of DC motor. In: 2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), St. Petersburg, pp. 998–1002 (2017). https://doi.org/10.1109/eiconrus.2017.7910725
10. Bolívar-Vincenty, C.G., Beauchamp-Báez, G.: Modelling the ball-and-beam system from newtonian mechanics and from lagrange methods. In: Twelfth LACCEI Latin American and Caribbean Conference for Engineering and Technology, 22–24 July 2014
11. Meenakshipriya, B., Kalpana, K.: Modelling and Control of Ball and Beam System using Coefficient Diagram Method (CDM) based PID controller. IFAC Proc. Volumes **47**(1), 620–626 (2014)
12. Tudoroiu, R.E., Ilias, N., Kecs, W., Casavela, S.V., Dobritoiu, M., Tudoroiu, N.: Real-time implementation of DC servomotor actuator with unknown uncertainty using a sliding mode observer. In: 2016 Federated Conference on Computer Science and Information Systems (FedCSIS), Gdansk, pp. 841–848 (2016)
13. Abdallaa, T.Y., Abed, A.A., Ahmed, A.A.: Mobile robot navigation using PSO-optimized fuzzy artificial potential field with fuzzy control. J. Intell. Fuzzy Syst. **32**(6), 3893–3908 (2017)
14. Gupta, M., Behera, L., Venkatesh, K.S.: PSO based modeling of Takagi-Sugeno fuzzy motion controller for dynamic object tracking with mobile platform. In: Proceedings of the International Multiconference on Computer Science and Information Technology, Wisla, pp. 37–43 (2010). https://doi.org/10.1109/imcsit.2010.5680034
15. Nasrinahar, A., Chuah, J.H.: Effective route planning of a mobile robot for static and dynamic obstacles with fuzzy logic. In: 2016 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), Batu Ferringhi, pp. 34–38 (2016). https://doi.org/10.1109/iccsce.2016.7893541
16. Parrilla, E., Riera, J., Torregrosa, J.R.: Fuzzy control for obstacle detection in object tracking. Math. Comput. Model. **52**(7–8), 1228–1236 (2010)
17. Masmoudi, M.S., Krichen, N., Masmoudi, M., Derbel, N.: Fuzzy logic controllers design for omnidirectional mobile robot navigation. Appl. Soft Comput. **49**(2016), 901–919 (2016)

Interactive Fuzzy Cellular Automata for Fast Person Re-Identification

Bahram Lavi(✉) and Muhammad Atta Othman Ahmed

Department of Electrical and Electronic Engineering, University of Cagliari,
Piazza d'Armi, 09123 Cagliari, Italy
{lavi.bahram,muhammad.ahmed}@diee.unica.it

Abstract. One of the goals of person re-identification systems is to support video-surveillance operators and forensic investigators to find an individual of interest in videos acquired by a network of non-overlapping cameras. This is attained by sorting images of previously observed individuals for decreasing values of their similarity with a given probe individual. Existing appearance descriptors, together with their similarity measures, are mostly aimed at improving ranking quality. We propose two fuzzy-based descriptors which are fast in terms of the processing time on descriptor generation and matching score computation. We then evaluate our approach on three benchmark data sets (VIPeR, i-LIDS, and ETHZ) with comparison of some descriptors in the state-of-the-art.

Keywords: Person re-identification · Fuzzy system
Cellular automata · Video surveillance system

1 Introduction

Person re-identification is a computer vision task consisting of recognizing an individual who had previously been observed over a network of cameras with non-overlapping fields of view [1]. One of its applications consists of supporting video surveillance operators and forensic investigators in retrieving all the videos showing an individual of interest, given an image of him/her as a query (*aka probe*). In this application scenario, the goal of a person re-identification system is returning to the user the frames or videos of all the individuals recorded by the camera network (*aka template gallery*) sorted for decreasing similarity to the probe, so that the user can find the occurrences of the individual of interest, ideally in the first positions. This task is challenging due to several issues typical of video surveillance footage, like low resolution, unconstrained pose, illumination changes, and occlusions, which do not allow to exploit strong biometrics like face. Clothing appearance is therefore one of the most widely used cues. Figure 1 illustrates some difficulties arising on person re-identification task.

Most of the existing techniques are based on defining a specific descriptor of clothing appearance (typically including color and texture), and a specific similarity measure between a pair of descriptors (evaluated as a *matching score*) which can be either manually defined or learnt from data [1,5,6,9,13].

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 147–157, 2018.
https://doi.org/10.1007/978-3-319-74690-6_15

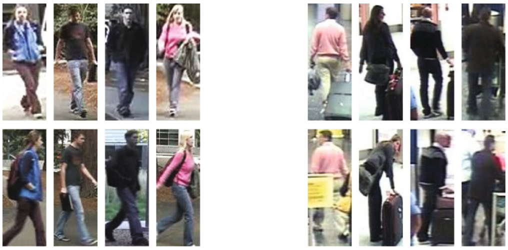
However, many of the existing similarity measures (either hand-crafted or learnt from data) are indeed rather complex, and require a relatively high processing time, e.g., [5,11,13]. On the other hand, in real-world applications the template gallery can be very large, and even if the processing time for a single matching score is low (e.g., the Euclidean distance between fixed-length feature vectors [13]), evaluating the matching scores for all the templates can be time-consuming. A known approach in the pattern recognition field, in particular for extracting texture information is to use Local Binary Pattern (LBP) [7]. However, this method is weak on handling shadow and illumination variation. Further, an improved version of LBP method, named scale invariant local ternary pattern (SILTP) [12], has been proposed with its promising results on illumination variation handling. Inspired by this method, in this paper we investigate whether and how to extract texture features by employing some fuzzy logic.

In this work we proposed two new fast descriptors for person re-identification task. First, we propose a method that consists of some pre-defined dominant colors by employing some fuzzy rules, named fuzzy dominant color (*hereinafter FDC*). Next, we propose another descriptor to distinguish the uniform color spaces from the feature space obtained by *FDC*. To this aim a fuzzy cellular automata is employed as a descirptor (*hereinafter FCA*). *FCA* is robust and simple due to parallel computing implementation [17]. However, despite from its simplicity, this method delivers a promising results when the image of an individual has a similar viewpoint on different camera views.

This paper is structured as follows. We first summarize related work in Sect. 2. In Sect. 3 we discuss our descriptors for person re-identification systems. In Sect. 4 we evaluate their effectiveness on three benchmark data sets, and compare them with some well-known descriptors on this area.

2 Background

The recent state-of-the-art abound descriptor for the task of person re-identification. These descriptors can be categorised by considering their cue on generating image signature; colour information [2,4], and many of existing works used as combination of different colour and texture information to help in attaining a better performance [5,13,16,25]. For person re-identification task, both of processing time and recognition rate are important to tackle in online applications. In this respect, some existing approaches are time consuming due to computation of matching scores for a target image against the templates in the gallery set. This may be caused of large number of feature elements generated as an image signature. Farenzena et al. [5] proposed symmetry driven accumulation of local features (SDALF) descriptor which subdivides body into four parts: left and right, torso and legs. Three kinds of features are extracted from each part: maximally stable color regions (MSCR), i.e., elliptical regions (blobs) exhibiting distinct color patterns (their number depends on the specific image); a weighted HSV color histogram (wHSV); and recurrent high-structured patches (RHSP) that characterize texture. A specific similarity measure is defined for each feature; the matching



(a) (b)

Fig. 1. (a) Sample image pairs from the VIPeR dataset [6] and (b) the i-LIDS dataset [24]. Each column represents a matching pair of individuals with upper and lower row in different camera view.

score is computed as their linear combination. Despite from the small features sizes for each of the descriptor, the generating such three descriptors are very time consuming. Another work proposed in [13] which is based on biologically-inspired features (BIF) obtained by Gabor filters with different scales over the HSV color channels. The resulting images are subdivided into overlapping regions; each region is represented by a covariance descriptor that encodes shape, location and color information. BIF and covariance descriptors are concatenated, and principle component analysis (PCA) technique is used to reduce its dimension. Moreover, in [11], they proposed a descriptor which extracts HSV histogram and two scales of the Scale Invariant Local Ternary Pattern histogram (characterizing texture) from overlapping windows; it then retains one only histogram from all windows at the same horizontal location, obtained as the maximum value among all the corresponding bins. These histograms are concatenated with the ones computed on a down-sampled image. The final image representation includes more than twenty five thousand elements for each image, which is very time consuming on matching score computation in real-world scenarios. A metric learning method is used to define the similarity measure.

On the other hand, in the context of fuzzy logic systems, some works have been proposed for the task of image processing [3,15,18,21,23] aim to image sharpening, filtering, image enhancement, and edge detection. Also, some fuzzy systems evaluated their method by using fuzzy cellular automata. One cellular automata (CA) proposed by [8], developed various CA algorithms for image enhancement. This work consists of algorithms of CA for improving, sharpening, and smoothing of an image. In [19], they proposed an algorithm of CA which exploits the genetic algorithm to discover the optimal CA rules in image edge detection problems. Another work in [14] proposed to use of a cellular learning automata (CLA) to improve accurate of detected edges and remove the weakened edges and noises. First the edges are detected by fuzzy logic, and then, the CLA is repeatably applied to them by considering the neighborhoods current state. In [21], proposed a texture histogram approach based on fuzzy cellular automata aiming to segment an image into a set of disjoint regions by concerning some uniform attributes. Sompong et al. [22] designed a framework to improve brain tumor segmentation. The proposed work by them is based on gray-level co-occurrence matrix which obtains by cellular automata.

To our knowledge, the issue of processing time has not been explicitly addressed so far except in [10], which the proposed approach aims to reduce the complexity and speed up of the other proposed descriptors in person re-identification task. In this paper, we address two new fast descriptors for person re-identification task in term of the processing time of a single descriptor generation and a matching score computation. However the recognition accuracy of our approach is similar to the descriptors in the literature and not outperform many of them, but the processing time is very low.

3 Proposed Approach

In standard person re-identification systems, $\mathbf{T}$ and $\mathbf{P}$ are the descriptors of a template and probe image, respectively, $m(\cdot,\cdot)$ the similarity measure between two descriptors, and $G = \{\mathbf{T}_1, \ldots, \mathbf{T}_n\}$ the template gallery. For a given $\mathbf{P}$, a standard re-identification system computes the matching scores $m(\mathbf{P}, \mathbf{T}_i)$, $i = 1, \ldots, n$, and returns the list of template images ranked for decreasing values of the score, and the most similar template is therefore ranked first. Ranking accuracy is typically evaluated using the cumulative matching characteristic (CMC) curve, i.e., the probability (recognition rate) that the correct identity is within the first ranks. In this paper, we propose an approach to construct the descriptor by using fuzzy system. First, we explain a descriptor which extracts the features based on some pre-defined dominant colors by using fuzzy logic (i.e. *FDC*), and then by using the first technique, we discuss on Fuzzy Cellular Automata (FCA) to construct a descriptor in order to describe a perceptually uniform color space for a given image. The whole scheme of the our proposed methods is presented at Fig. 2.

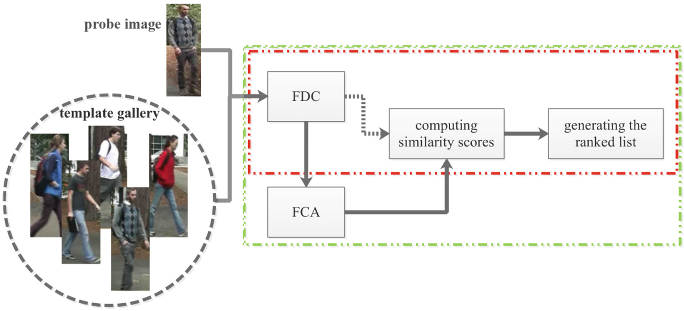
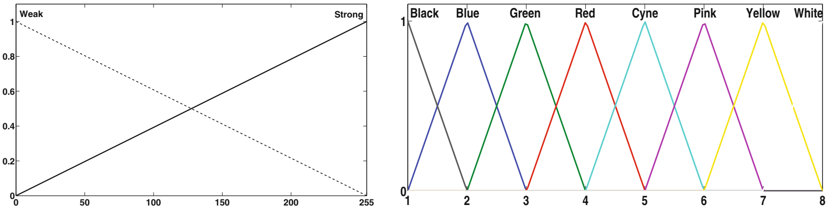


Fig. 2. The whole scheme of the proposed FDC and FCA techniques for person re-identification task. Note that if one employs FCA as the main descriptor, the obtained features by FDC are instead left to FCA descriptor, and then the similarity scores are computed.

3.1 Fuzzy Dominant Color Descriptor

Fuzzy dominant color descriptor (FDC) consists of extracting color information using a number of pre-defined dominant colors in RGB color space. The number of dominant colors is depended on the number of membership function (MF), defined for each RGB channel. Pre-defined dominant colors are evaluated by using some fuzzy rules. This allows us to reduce the huge number of possible colors in the RGB color space, which can be equal to 256^3, into the limited number of colors. For any acquired images in a network camera, dominant colors might be different, by passing a camera to another, due to illumination and brightness. Thus, computing dominant color of an image with the common way, may achieved a different dominant color for that specific image. At the following we explain the process defining the dominant color by fuzzy system.

To construct such descriptor, first, MFs are defined for each color channel in RGB space for the valid pixel values (MF of each channel is denoted by α, β, γ for *red*, *green*, and *blue* channels, respectively) in the range of [0 255] as input space, which are mapped to a membership values in the range of [0 1]. For instance, let $n = 2$, where n is the number of MFs (e.g. namely may define as *Weak* and *Strong*) defined for each RGB color channel (see Fig. 3(a)). Moreover, output MFs are defined in which are the expected dominant colors (see Fig. 3(b)). Then, fuzzy rules are defined which describe those membership functions. In this manner, the fuzzy rules describe the dominant colors. The number of fuzzy rules is depended on the number of MFs, where in our proposed method is $R = n^3$, where R is the number of fuzzy rules. However, the number of dominant colors described by fuzzy rules is equal R (see Table 1).



(a) (b)

Fig. 3. (a) Fuzzy input membership function for a specific color channel when 8 fuzzy dominant colours are defined. (b) Fuzzy output membership function for 8 pre-defined dominant colours.

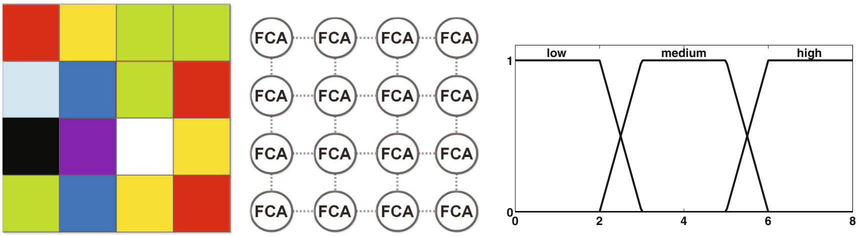
To apply the described fuzzy system for a given image, let I be a given image, and $I_{c \in RGB}(x, y)$ be a set of three pixels value corresponded at each RGB channel. The corresponded fuzzy dominant color for a given triple pixels can be computed as a function of fuzzy system $D(x, y) = F(I_r, I_g, I_b)$. The obtained values on D demonstrate the expected dominant colors of each pixel for the given image. Thereafter, a histogram for D is computed, and represented as an image signature. Denote that the final image signature (D) always equals to R.

Table 1. Fuzzy rules for defining 8 dominant colors.

If	Then color
α Weak *and* β Weak *and* γ Weak	*Black*
α Strong *and* β Weak *and* γ Weak	*Red*
α Weak *and* β Strong *and* γ Weak	*Green*
α Weak *and* β Weak *and* γ Strong	*Blue*
α Strong *and* β Weak *and* γ Strong	*Pink*
α Strong *and* β Strong *and* γ Weak	*Yellow*
α Weak *and* β Strong *and* γ Strong	*Cyne*
α Strong *and* β Strong *and* γ Strong	*White*

3.2 Fuzzy Cellular Automata Descriptor

FDC descriptor, as explained at sect. 3.1, could not be described the given images well. However, the color information is the most interesting cue for image processing for some kind of images which neither has no specific shape nor easy to extract the texture information. In this manner, we aim to compute the image signature by considering the state of the neighbours of each computed pixel's dominant colors in matrix D obtained by *FDC*. In fact, our goal is to achieve the uniform information of the dominant colors of the image. To obtain such image representation, we employ Fuzzy Cellular Automata (*FDA*), which is applied for the matrix D. However, *FDA* can be defined as quadruple of $\{D, Q, r, f\}$, where D, Q, r, and f are finite non-empty matrix of computed dominant colours by *FDC*, a set of cells state, r is the neighbourhood radius, and fuzzy transferring function which defines the state of a specific cell. Note that the size of matrix Q must be as same size as matrix of D, and updated subsequently during the process (see Fig. 4). Also note that each cell is influenced by the current state of the neighbourhoods cell. Take also into account that the neighbourhood radius has been chosen as $r = 8$. The state of each cell may have the one value in a set of



(a) (b) (c)

Fig. 4. (a) An example of computed dominant colours using *FDC* ($D_{4\times4}$), (b) shows a set of *FCA* ($Q_{4\times4}$), and (c) illustrates an example of *FCA* input membership function when $r = 8$

$states = \{Low, Medium, Strong\}$ which shows the intensity of the neighbours of each dominant colors in D. For a given $D(x, y)$ and its neighbour values as the input of transition function, and then calculates the state of the cell in $Q(x, y)$. Once the values in Q obtained, then it is easy to compute the histogram for computed matrix of Q, and finally the result is represented as the image representation. Note that the final feature size would be $R * 3$ caused by three different states, $\{Low, Medium, Strong\}$ have been taken into account. (see an example of final image signature on Fig. 5).

L1	M1	H1	L2	M2	H2	L3	M3	H3	L4	M4	H4	L5	M5	H5	L6	M6	H6	L7	M7	H7	L8	M8	H8
0	0	0	20	10	0	24	191	1171	117	343	703	24	114	146	0	0	0	0	0	0	0	0	0

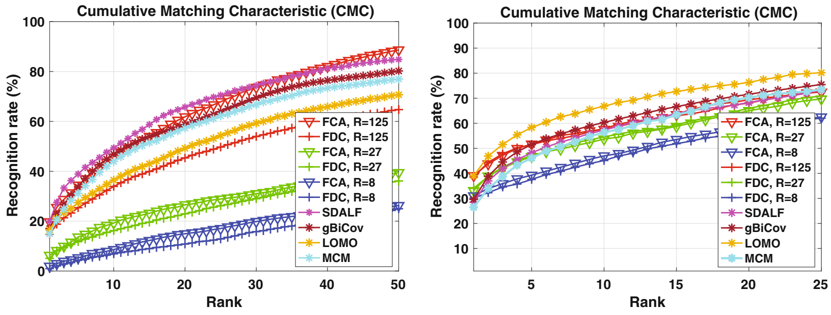
Fig. 5. An example of generated image signature by using FCA approach. On this, 8 dominant colors have been defined by fuzzy system.

4 Experimental Results

We compared our descriptors with four well-known descriptors in person re-identification problem: SDALF [5], gBiCov [13], MCM [20], and LOMO [11]. We carried out our experimental evaluations on three benchmark data sets: VIPeR, i-LIDS and ETHZ. VIPeR [6] is a challenging data set for person re-identification; it is made up of two images of 632 individuals from two camera views, with pose and illumination changes, cropped and scaled to 128 × 48 pixels. i-LIDS contains 476 images of 119 pedestrians taken at an airport hall from non-overlapping cameras, with pose and lightning variations and strong occlusions. ETHZ contains three video sequences of a crowded street from two moving cameras; images differ in size and exhibit illumination changes, scale variations, and occlusions. The sub-sequences are structured as follow: SEQ. #1 contains 83 pedestrians (4.857 images), SEQ. # 2 contains 35 pedestrians (1.961 images), and SEQ. #3 contains 28 pedestrians (1.762 images). We rescaled the images of i-LIDS and ETHZ to the same size of 128 × 48 pixels as in VIPeR, to get a similar processing time.

4.1 Experimental Setup

One image for each person was randomly selected to build the template gallery; the other images formed the probe gallery. As in [5], for each data set we repeated our experiments on ten different subsets of individuals, using one image of each individual as template and one as probe, and reported the average CMC curve over the ten runs. We used an Intel Core i5 2.6 GHz CPU. For computing the matching scores of generated descriptors by *FDC* and *FCA*, we employed Bhattacharya distance for computing the similarity for each pair of individuals (i.e. probe and template), which is more suitable distance metric for independent features (Fig. 6).



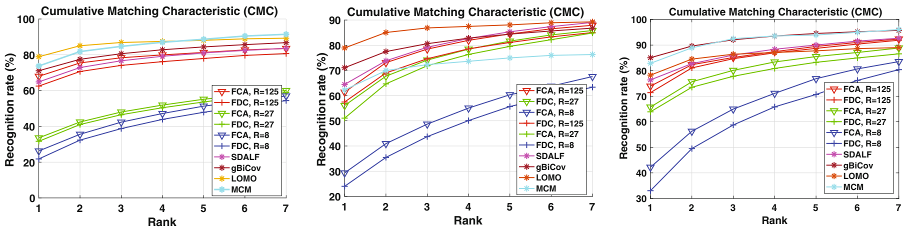
(a) VIPeR

(b) i-LIDS

Fig. 6. CMC curves attained by *FDC* and *FCA* on the two datasets used in the experiments.

4.2 Results

We evaluated our experimental results using *FDC* for dominant colors in $R = \{8, 27, 125\}$. We then carried out the *FCA* descriptor on the generated set of D obtained by *FDC*. However, our descriptors were not able to outperform the state-of-the-art descriptors on VIPeR and ETHZ datasets (because of viewpoint change and occlusion) but the results are quite close those descriptors when $R = 125$. This only outperformed on i-LIDS data set when $R = \{27, 125\}$. This is hence why that many of the individuals have a similar viewpoint on i-LIDS data sets (see Fig. 7). Despite of the recognition rate, our descriptors were faster than the others in terms of the processing time of matching scores



(a) ETHZ1

(b) ETHZ2

(c) ETHZ3

Fig. 7. CMC curves attained by *FDC* and *FCA* on the ETHZ dataset used in the experiments.

and descriptor generation. Regarding to the processing time, we analyse also the processing efficiency of our method with some descriptors in the literature. The average processing time for computing a single matching score as well as a single descriptor generation are reported in Table 2.

Table 2. Average processing time (in sec.) for computing one matching score (t_M) and one descriptor generation (t_D), for each of the two descriptors *FCA* and *FCA* and comparing them with descriptors in the literature. Moreover, the relevant feature size obtained by each descriptor is reported and indicated by F_s.

Descriptor	R	t_M	t_D	F_s
FCA	8	0.026	0.00016	24
	27	0.028	0.00027	81
	125	0.034	0.0067	375
FDC	8	0.023	0.0001	8
	27	0.027	0.00012	27
	125	0.029	0.0089	125
SDALF		9.4	1.896	Variant
gBiCov		0.04	1.34	≈6000
LOMO		0.037	0.08	≈27000
MCM		27.4	3.2	Variant

5 Conclusions

We proposed two fast descriptors for person re-identification task. These descriptors aimed to tackle the appearance of the individuals to find the uniform color space of the images by using some fuzzy logic. Our approach is inspired by the well-known scale invariant local ternary pattern (SILTP) approach used in pattern recognition systems which we adopted the fuzzy logic problem to extract some texture information to handle illumination and shadow. In practical settings, the attainable ranking accuracy can be improved by defining more the different pre-defined colors and fuzzy rules. The experimental results evidenced on three benchmark VIPeR, i-LIDS, and ETHZ data sets, showed the proposed approach has a very similar recognition rate and outperform only on i-LIDS, with respect to the state-of-the-art descriptors, in very low computational cost.

References

1. Bedagkar-Gala, A., Shah, S.K.: A survey of approaches and trends in person re-identification. Image Vis. Comput. **32**(4), 270–286 (2014)
2. Cheng, D.S., Cristani, M., Stoppa, M., Bazzani, L., Murino, V.: Custom pictorial structures for re-identification. In: BMVC, p. 6 (2011)

3. Choi, Y., Krishnapuram, R.: A robust approach to image enhancement based on fuzzy logic. IEEE Trans. Image Proc. **6**(6), 808–825 (1997)
4. Dikmen, M., Akbas, E., Huang, T.S., Ahuja, N.: Pedestrian recognition with a learned metric. In: ACCV 2010, pp. 501–512. Springer, Heidelberg (2011)
5. Farenzena, M., Bazzani, L., Perina, A., Murino, V., Cristani, M.: Person re-identification by symmetry-driven accumulation of local features. In: 2010 IEEE Conference on CVPR, pp. 2360–2367. IEEE (2010)
6. Gray, D., Tao, H.: Viewpoint invariant pedestrian recognition with an ensemble of localized features. In: ECCV 2008, pp. 262–275. Springer, Heidelberg (2008)
7. Guo, Z., Zhang, L., Zhang, D.: A completed modeling of local binary pattern operator for texture classification. IEEE Trans. Image Proc. **19**(6), 1657–1663 (2010)
8. Hernandez, G., Herrmann, H.J.: Cellular automata for elementary image enhancement. Graph. Models Image Proc. **58**(1), 82–89 (1996)
9. Hirzer, M., Roth, P.M., Bischof, H.: Person re-identification by efficient impostor-based metric learning. In: AVSS, pp. 203–208. IEEE (2012)
10. Lavi, B., Fumera, G., Roli, F.: A multi-stage approach for fast person re-identification. In: Proceedings of the Structural, Syntactic, and Statistical Pattern Recognition: Joint IAPR International Workshop, S+SSPR, pp. 63–73. Springer, Cham (2016)
11. Liao, S., Hu, Y., Zhu, X., Li, S.Z.: Person re-identification by local maximal occurrence representation and metric learning. In: CVPR, pp. 2197–2206 (2015)
12. Liao, S., Zhao, G., Kellokumpu, V., Pietikäinen, M., Li, S.Z.: Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes. In: IEEE Conference on CVPR, pp. 1301–1306. IEEE (2010)
13. Ma, B., Su, Y., Jurie, F.: Covariance descriptor based on bio-inspired features for person re-identification and face verification. Image Vis. Comput. **32**(6), 379–390 (2014)
14. Patel, D.K., More, S.A.: Edge detection technique by fuzzy logic and cellular learning automata using fuzzy image processing. In: 2013 International Conference on Computer Communication and Informatics (ICCCI), pp. 1–6. IEEE (2013)
15. Pradipta, M., Chaudhuri, P.P.: Fuzzy cellular automata for modeling pattern classifier. IEICE Trans. Inf. Syst. **88**(4), 691–702 (2005)
16. Prosser, B., Zheng, W.S., Gong, S., Xiang, T., Mary, Q.: Person re-identification by support vector ranking. In: BMVC, p. 6 (2010)
17. Rosin, P.L.: Image processing using 3-state cellular automata. Comput. Vis. Image Underst. **114**(7), 790–802 (2010)
18. Russo, F., Ramponi, G.: A fuzzy filter for images by impulse noise. IEEE Signal Process. Lett. **3**(6), 168–170 (1996)
19. Sahota, P., Daemi, M., Elliman, D.: Training genetically evolving cellular automata for image processing. In: Speech, Image Processing and Neural Networks, pp. 753–756. IEEE (1994)
20. Satta, R., Fumera, G., Roli, F., Cristani, M., Murino, V.: A multiple component matching framework for person re-id. In: ICIAP, pp. 140–149. Springer, Heidelberg (2011)
21. Shahverdi, R., Tavana, M., Ebrahimnejad, A., Zahedi, K., Omranpour, H.: An improved method for edge detection and image segmentation using fuzzy cellular automata. Cybern. Syst. **47**(3), 161–179 (2016)
22. Sompong, C., Wongthanavasu, S.: An efficient brain tumor segmentation based on cellular automata and improved tumor-cut algorithm. Expert Syst. Appl. **72**, 231–244 (2017)

23. Tyan, C.Y., Wang, P.P.: Image processing-enhancement, filtering and edge detection using the fuzzy logic approach. In: Second IEEE International Conference on Fuzzy Systems, pp. 600–605. IEEE (1993)
24. Zheng, W.S., Gong, S., Xiang, T.: Associating groups of people. In: BMVC, vol. 2, p. 6 (2009)
25. Zheng, W.S., Gong, S., Xiang, T.: Reidentification by relative distance comparison. In: IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 653–668 (2013)

Decision Support System for Determination of Forces Applied in Orthodontic Based on Fuzzy Logic

Lamia Nabil Omran[1,3], Kadry Ali Ezzat[1,3(✉)], and Aboul Ella Hassanien[2,3]

[1] Biomedical Engineering Department, Higher technological Institute,
10th of Ramadan, Egypt
[2] Faculty of Computers and Information, Cairo University, Cairo, Egypt
[3] Scientific Research Group in Egypt (SRGE), Cairo, Egypt
Kadry_ezat@hotmail.com
http://www.egyptscience.net

Abstract. The aim of this paper is to design a decision support system based on fuzzy logic to assist the dentist in teeth alignment. There is lack of consistency among dentists in choosing the treatment plan. Moreover, there is no decision support system currently available to verify and support such decision making in dentistry. This paper presents a decision support system logic for determination of forces applied in an Orthodontic system based on fuzzy logic. We designed a knowledge base with 27 rules and used Mamdani inference algorithm to decide the possible the suitable force required to be applied on the archwire and suggest to decides for operation continuity or not to the dentist. It suggests suitable values of displacements, Young's modulus and degree of pain as inputs that effect on the forces and decision of continuity as outputs. The results show Young's modulus of archwires material plays an important role in detecting the value of force required to be applied on the archwire during the treatment process with consideration of the displacement to be moved by the tooth. Also, the result shows that the degree of patients pain is an important input in the decision of continuing treatment plan or not.

Keywords: Expert systems · Fuzzy logic · Orthodontic system
Decision support system

1 Introduction

Fuzzy logic is based upon fuzzy set theory. A fuzzy set is an expansion of the classical variable set between and including 0 and 1 [7]. A membership function is used to define how each element of the input space is assigned a value between 0 and 1. To evaluate a fuzzy system inference is then utilized [8]. A fuzzy inference system is a framework that simulates the behavior of a given system using IF-THEN rules and is based on expert knowledge or variable data on the system [9]. Rules are statements of knowledge that relate the computability of

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 158–168, 2018.
https://doi.org/10.1007/978-3-319-74690-6_16

fuzzy premise propositions to one or more fuzzy spaces [10]. White [11] identified more than three decision support systems. This system is divided into seven subareas of dentistry. White also classified the system according to the knowledge representation used, including algorithmic, statistical, rule-based, and image-processing systems. Brickley et al. [12] developed a system to make lower-third molar treatment-planning decisions, using a software-based fuzzy logic. This study demonstrates that it is possible to use fuzzy logic to provide reliable decision support for lower-third-molar treatment planning. Suebnukarn et al. [13] presented a decision support model that describes the mutual relationships among multiple variables for assessment of the outcome of orthodontic treatment. The aim behind designing this system was to offer support to dental specialists while selecting the values of forces required in teeth alignment depending on the suitable displacements of each tooth and Youngs modulus of archwire material. This system also helps the dentist in deciding to continue the treatment or not according to the degree of patient's pain Deciding on a treatment plan is a challenging task.

Expert Systems are computer programs that are derived from a branch of software engineering research called Artificial Intelligence (AI) [1]. The term intelligence covers numerous cognitive skills, including the ability of problem-solving, learn, and understand language [2]. AI's scientific objective is to understand intelligence by building computer programs that exhibit intelligent behavior [3]. It is concerned with the ideas and techniques of symbolic inference, or reasoning, by a computer, and how the knowledge is used to make those inferences will be represented inside the machine. Expert systems are knowledge-based computer programs designed to provide assistance in diagnosis and treatment planning [4]. They assist the practitioner in decision-making. Fuzzy logic is a multi-valued logic, which allows the evaluation of a set of variables by defining intermediate values between the conventional evaluation schemes such as true and false [5]. It essentially enables computers a more human-like way of thinking. It requires the definition of fuzzy variables sets extracted from the physical problem [6].

2 The Proposed Orthodontic System

For the proposed system, the pain, Youngs modulus and given displacements are considered as inputs, while force and decision are as output. The resultant may be light, medium or hard treatments [14]. The FLS has four fundamental useful parts: Fuzzier, Inferences Engine, Defuzzier and Knowledges Base that is made out of if-then standards as in Fig. 1.

2.1 Fuzzification

Fuzzification implies creating of the functions for variables membership [15]. The functions for membership significance values for every variable were in the

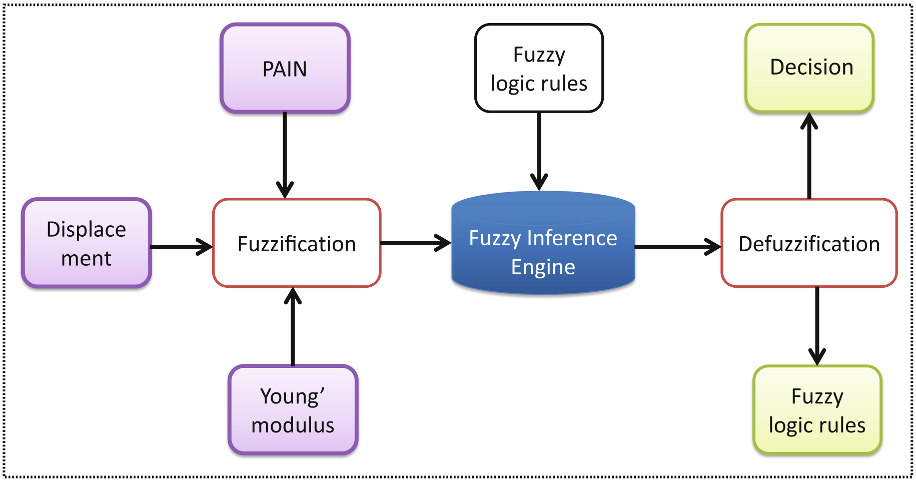


Fig. 1. The general architecture of the proposed orthodontic system

range from 0 to 1. The determination was depend upon three variables: Low-pull, Medium-pull and High-pull for Displacement, three other variables: Mild, Moderates and Severe for pain, three variables: Excellent (low), good (medium) and bad (high) for Youngs modulus, while for the outputs three variables: low force, medium force and high force for force and two variables stop, continue for decision.

2.2 Inputs for Fuzzy Logic System

- **Displacement:** For Displacement value (x) with range [0.1 2.5], the fuzzy membership terms, utilizing trapezoidal (trapmf) membership functions for variables of Low-pull, Medium-pull and High-pull respectively. Where *Low-pull:* The low pull membership function is given by Eq. (1).

$$f(x,0,0.1,0.5,0.8) = \begin{cases} 0, & x \leq 0 \\ \frac{x-0}{0.1-0}, & 0 \leq x \leq 0.1 \\ 1 & 0.1 \leq x \leq 0.5 \\ \frac{0.8-x}{0.8-0.5}, & 0.5 \leq x \leq 0.8 \\ 0, & 0.8 \leq x \end{cases} \tag{1}$$

Medium-pull: The medium pull membership function are given by Eq. (2).

$$f(x,0.5,1,1.35,1.73) = \begin{cases} 0, & x \leq 0.5 \\ \frac{x-0.5}{1-0.5}, & 0.5 \leq x \leq 1 \\ 1 & 1 \leq x \leq 1.35 \\ \frac{1.73-x}{1.73-1.35}, & 1.35 \leq x \leq 1.73 \\ 0 & 1.73 \leq x \end{cases} \tag{2}$$

High-pull: The high pull membership function is given by Eq. (3).

$$f(x, 1.45, 1.6, 2.5, 2.6) = \begin{cases} 0, & x \leq 1.45 \\ \frac{x-1.45}{1.6-1.45}, & 1.45 \leq x \leq 1.6 \\ 1 & 1.6 \leq x \leq 2.5 \\ \frac{2.6-x}{2.6-2.5}, & 2.5 \leq x \leq 2.6 \\ 0, & 2.6 \leq x \end{cases} \tag{3}$$

- **PAIN:** For Pain valued (x) with range [0 1], the membership termed, utilizing Z-shaped, Gaussian and Si-shaped membership functions for variables Mild, Moderate and Severe respectively are:
  Mild: The Mild membership function is given by Eq. (4).

$$f(x, 0.1, 0.4) = \begin{cases} 1, & x \leq 0.1 \\ 1 - 2(\frac{x-0.1}{0.4-0.1})^2, & 0.1 \leq x \leq \frac{0.1+0.4}{2} \\ 2(\frac{x-0.4}{0.4-0.1})^2 & \frac{0.1+0.4}{2} \leq x \leq 0.4 \\ 0 & 0.4 \leq x \end{cases} \tag{4}$$

  Moderate: The Moderate membership function is given by Eq. (5).

$$f(x, 0.1699, 0.5) = e^{\frac{-(x-0.5)^2}{2*0.1699^2}} \tag{5}$$

  Severe: The severe membership function are given by Eq. (6).

$$f(x, 0.6, 0.9) = \begin{cases} 0, & x \leq 0.6 \\ 2(\frac{x-0.6}{0.3})^2, & 0.6 \leq x \leq \frac{0.6+0.9}{2} \\ 1 - 2(\frac{x-0.9}{0.3})^2 & \frac{0.6+0.9}{2} \leq x \leq 0.9 \\ 0 & 0.9 \leq x \end{cases} \tag{6}$$

- **Young modulus:** For Youngs modulus valued (x) with range [60 220], the membership termed, using Pimf membership functions for variables Excellent (low), good (medium) and bad (high) respectively. Where
  Excellent (low): The Excellent (low) membership function is given by Eq. (7).

$$f(x, 55, 60, 110, 135) = \begin{cases} 0, & x \leq 55 \\ 2(\frac{x-55}{60-55})^2, & 55 \leq x \leq \frac{55+60}{2} \\ 1 - 2(\frac{x-55}{60-55})^2 & \frac{55+60}{2} \leq x \leq 60 \\ 1 & 60 \leq x \leq 110 \\ 1 - 2(\frac{x-110}{135-110})^2, & 110 \leq x \leq \frac{110+135}{2} \\ 2(\frac{x-110}{135-110})^2, & \frac{110+135}{2} x \leq 135 \\ 0 & 135 \leq x \end{cases} \tag{7}$$

Good (medium): The Good (medium) membership function is given by Eq. (8).

$$f(x,110,120,180,190)=\begin{cases}0, & x\le 110\\ 2(\frac{x-110}{120-110})^2, & 110\le x\le \frac{110+120}{2}\\ 1-2(\frac{x-110}{120-110})^2, & \frac{110+120}{2}\le x\le 120\\ 1 & 120\le x\le 180\\ 1-2(\frac{x-180}{190-180})^2, & 180\le x\le \frac{180+190}{2}\\ 2(\frac{x-180}{190-180})^2, & \frac{180+190}{2}x\le 190\\ 0 & 190\le x\end{cases} \tag{8}$$

Bad (high): The bad (high) membership function is given by Eq. (9).

$$f(x,170,200,220,225)=\begin{cases}0, & x\le 165\\ 2(\frac{x-165}{200-165})^2, & 165\le x\le \frac{165+200}{2}\\ 1-2(\frac{x-165}{200-165})^2 & \frac{165+200}{2}\le x\le 200\\ 1 & 200\le x\le 220\\ 1-2(\frac{x-220}{225-220})^2 & 220\le x\le \frac{220+225}{2}\\ 2(\frac{x-200}{225-200})^2, & \frac{220+225}{2}x\le 225\\ 0 & 225\le x\end{cases} \tag{9}$$

2.3 Outputs Variables of Fuzzy Logic System

- **Forces values:** For force value (x) with range [0.1 5], the fuzzy membership termed, using trapezoidal (trapmf) membership functions for variables low force, medium force and high force respectively. Where
  Low force: The low force membership function are given by Eq. (10).

$$f(x,0,0.1,0.7,1.2)=\begin{cases}0, & x\le 0\\ \frac{x-0}{0.1-0}, & 0\le x\le 0.1\\ 1 & 0.1\le x\le 0.7\\ \frac{1.2-x}{1.2-0.7}, & 0.7\le x\le 1.2\\ 0 & 1.2\le x\end{cases} \tag{10}$$

Medium force: The medium force membership function are given by Eq. (11).

$$f(x,0.7,1.5,2.6,3)=\begin{cases}0, & x\le 0.7\\ \frac{x-0.7}{1.5-0.7}, & 0.7\le x\le 1.5\\ 1, & 0.5\le x\le 2.6\\ \frac{3-x}{3-2.6}, & 2.6\le x\le 3\\ 0, & 3\le x\end{cases} \tag{11}$$

High force: The high force membership function are given by Eq. (12).

$$f(x,2.5,3.4,5,5.1)=\begin{cases}0, & x\le 2.5\\ \frac{x-2.5}{3.4-2.5}, & 2.5\le x\le 3.4\\ 1, & 3.4\le x\le 5\\ \frac{5.1-x}{5.1-5}, & 5\le x\le 5.1\\ 0, & 5.1\le x\end{cases} \tag{12}$$

- Decision: For Decision value (x) with range [0 1], the fuzzy membership termed, using Z-shaped membership functions for variable continue, and S-shaped function for variable stop are:

  Continue: The continue membership function is given by Eq. (13).

$$f(x,0,0,6) = \begin{cases} 1, & x \leq 0 \\ 1 - 2(\frac{x-0}{0.6-0})^2, & 0 \leq x \leq \frac{0+0.6}{2} \\ 2(\frac{x-0.6}{0.6-0})^2, & \frac{0+0.6}{2} \leq x \leq 0.6 \\ 0, & 0.6 \leq x \end{cases} \tag{13}$$

  Stop: The stop membership function is given by Eq. (14).

$$f(x,0,6,1) = \begin{cases} 0, & x \leq 0.5 \\ 2(\frac{x-0.5}{1-0.5})^2, & 0.5 \leq x \leq \frac{0.5+1}{2} \\ 1 - 2(\frac{x-1}{1-0.5})^2, & \frac{0.5+1}{2} \leq x \leq 1 \\ 1, & 1 \leq x \end{cases} \tag{14}$$

2.4 Rules for Fuzzy Logic system

The output process of the presented system based on the construction of if-then fuzzy rules. Three inputs parameters, The Bain, youngs modulus of arch wire and given displacements each of three variables each thus, 27 (9×3) rule combinations formed. These rules are described in the rule editor of FIS supported by MATLAB.

2.5 Fuzzy Inference Engine

The Mamdani method which configure the supervision properties for the given system into its controlled properties [16,17].

2.6 Defuzzification

It converts the fuzzy sets acquired by the inferences engines into detected values. The output of the system defuzzified utilizing centre of Area defuzzification technique, which also known as the Centre of Gravity (CoG) technique, the controller of fuzzy computes the area under the standardized membership functions and with respect to the output variable range. After that the fuzzy logic controller utilizes in Eq. (15) to compute center of that area [18].

$$CoA = \frac{\int_{x_{min}}^{x_{max}} f(x) * x dx}{\int_{x_{min}}^{x_{max}} f(x) dx} \tag{15}$$

Where CoA is centre of area, x is the variable value, and x_{min} and x_{max} represent the range. This method is most effective method that computes the perfect normalization between varied output terms.

3 Results and Discussion

Figure 2 illustrates the relation between any two factors on the x-y planes and resulting output on the z plan. The color code is displayed by Fig. 2a. The relation between two inputs displacement and youngs modulus is equally affecting the output (force) by the pattern displayed in Fig. 2b, the relation between required force and both displacement and youngs modulus is linear (raising gradually) until a certain point, then the force is constant for any displacement in the range of youngs modulus (60–100 GPa) but for higher youngs modulus (100–160 GPa) the force react in 3 different patterns. It increases gradually for displacement (0–1 mm) then raise again for range (1–1.5 mm) then raise again for range (1.5–2.5 mm) to a constant value of force (4 N). For the youngs modulus of range from (160–200 GPa) and with very low (neglected) displacements (0–0.5 mm) the forces increase gradually from (0–2 N) which may harm the teeth, while with higher displacements (0.5–1.5 mm) the forces increases gradually from (2–4 N) then the force raise again to a constant value (4 N) for displacements from (2–2.5 mm).

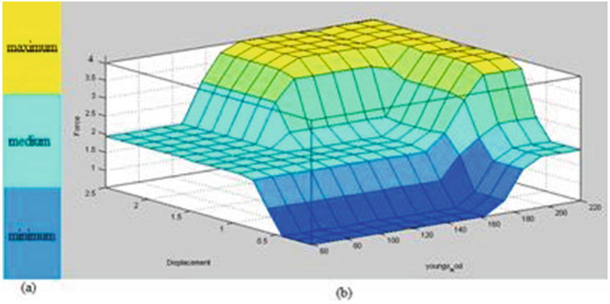


Fig. 2. (a) the reference chart for output surface, (b) the relation between inputs (Displacement, youngs modulus) and force

The relation between the two inputs pain, youngs modulus as shown in Fig. 3 is not linked, as the output force depend on the youngs modulus more than pain but if the patient feel with severe pain, the force must be reduced. For low displacement (0.1 mm) the force is constant along the pain with youngs modulus (60–100 GPa), while the force increase gradually with youngs modulus of range (100–180 GPa) and with low pain (0–0.3). The force is constant at the range of youngs modulus (100–140 GPa) and pain from (0.4–1), then the force increase gradually for youngs modulus of range (160–220 GPa) with pain of range (0.4–1). The force range here is in low range (0–1.8 N).

The relation between the two inputs pain, displacement as shown in Fig. 4 is not strongly linked, as the output force depend on the displacement more than pain but if the patient feel with severe pain, the force must be reduced. The force is constant for displacements of range (0–0.5 mm) and pain of range from

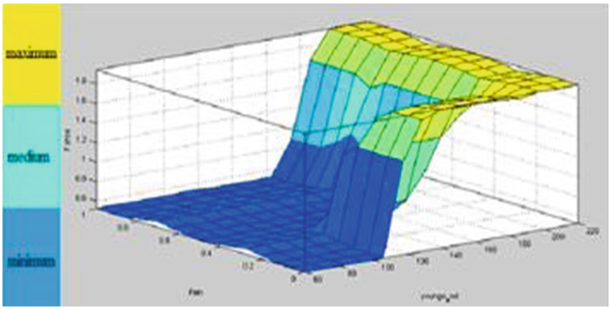


Fig. 3. The relationship between inputs (pain, youngs modulus) and (force)

(0–1) then the force increase gradually for displacements of range (0.5–1 mm) and pain of range from (0–1). The force become constant at the displacement of range (1–1.5) and of pain increased within range between (0.3–1), finally the forces increase gradually with displacement of range between (1.5–2.5 mm) and with low pain in range of (0–0.2).

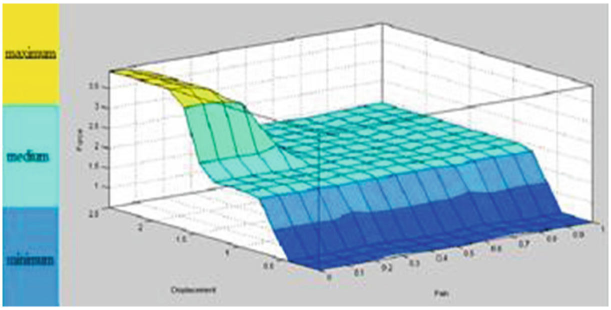


Fig. 4. The relation between inputs (pain, displacement) and output (force)

The two inputs displacement and youngs modulus not effect on the decision of continuing the orthodontic treatment so if the pain is mild or moderate and the displacement, youngs modulus are variate within their ranges (0–2.5 mm) and (60–120 GPa) correspondingly the decision will be continue (0–0.5). As shown in Fig. 5, it is clear that the maximum value of decision is 0.2 which is fall in the range of continuity of the treatment and vice versa will occurred when the pain is severe.

The input pain play an important role in the decision making of the treatment plan more than youngs modulus and displacement as shown in Figs. 6 and 7, so that if the pain was mild (0–0.4) or moderate (0.4–0.6), and the displacement, youngs modulus are variety within their ranges (0–2.5 mm) and (60–120 GPa) correspondingly the decision will be continue (0–0.5) but if the pain was severe (0.6–1) the decision will stop (0.5–1) the treatment plan and reduce the force.

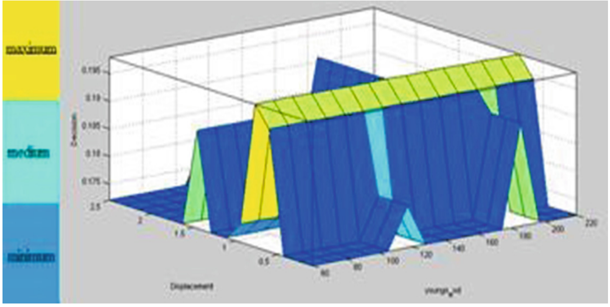


Fig. 5. The relation between inputs (youngs modulus, displacement) and output (decision)

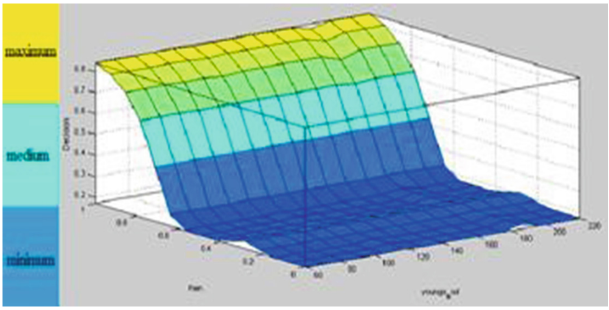


Fig. 6. The relation between inputs (youngs modulus, pain) and output decision

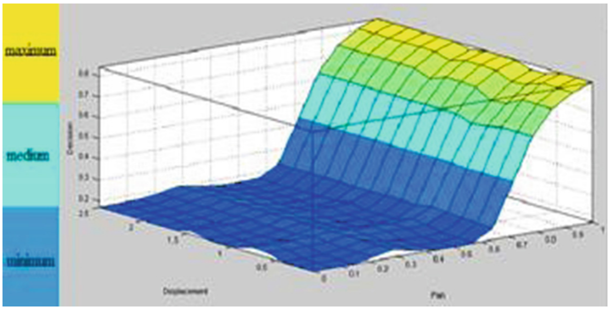


Fig. 7. The relation between inputs (displacement, pain) and output (decision)

4 Conclusions

In this paper, fuzzy logic provides effective tools for modeling uncertainty in human reasoning. A fuzzy inference system represents knowledge in IF-THEN rules and implements fuzzy reasoning. The expert system shows the effect of the size and the materials of the archwires on the displacements of teeth and the forces that required causing this displacement in addition to the suggestion of reducing the applied forces if the patient feels with severe pain. Most of the previous expert systems were depend on fuzzy logic tools only without depending on any other tools which make those expert systems not integrated in addition to as they didnt take youngs modulus, displacement and pain in consideration at the same time and they didnt introduce treatment plan suggestions. Brickley developed a system to make only one tooth (lower third molar) treatment-planning decisions, using a software based fuzzy logic. Vijay Kumar introduces an expert system based on fuzzy logic tool only for treatment of mobile tooth; it depended on insufficient inputs neglecting displacements and types of the arch wires, which were important inputs in our expert system. The output produced by or predicted by those systems was slightly homogeneous to dentists prediction.

References

1. Mirza, M., Gholam Hosseini, H., Harrison, M.J.: A fuzzy logic-based system for anaesthesia monitoring. In: Proceedings of Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC, Buenos Aires, pp. 3974–3977 (2010)
2. Allahverdi, N., Akcan, T.A.: Fuzzy expert system design for diagnosis of periodontal dental disease. IEEE (2011)
3. Su, X., Xia, F., Wu, L., Philip Chen, C.L.: Event-triggered fault detector and controller coordinated design of fuzzy systems. IEEE Trans. Fuzzy Syst. **PP**(99), 1 (2017)
4. Xue, B., Zhang, M., Browne, W.N.: Particle swarm optimization for feature selection in classification: a multi-objective approach. IEEE Trans. Cybern. **43**(6), 1656–1671 (2013)
5. Gader, P., Keller, J.M., Cai, J.: A fuzzy logic system for the detection and recognition of handwritten street numbers. IEEE Trans. Fuzzy Syst. **3**(1), 83–95 (1995)
6. Allahverdi, N.: Some applications of fuzzy logic in medical area. In: Proceedings on the 3rd International Conference on Application of Information and Communication Technologies (AICT 2009), 14–16 October 2009, Azerbaijan, Baku. IEEE (2009)
7. Soleymani, S.A., Abdullah, A.H., Zareei, M.: A secure trust model based on fuzzy logic in vehicular ad hoc networks with fog computing. IEEE Access **5**, 15619–15629 (2017)
8. Dewi, D.P.S.: Sistem Pakar Diagnosa Penyakit Jantung dan Paru dengan Fuzzy Logic dan Certainty Factor. Merpati, vol. 2, No. 3 (2014)
9. Nmeth, B., Laboncz, S., Kiss, I., Cspes, G.: Transformer condition analyzing expert system using fuzzy neural system. In: IEEE International Symposium on Electrical Insulation (ISEI), Canada (2010)
10. Hong, G., Chen, X., Xue, X., Zhang, S.: Expert systems for fault diagnosis integrating neural network and fuzzy inference. In: International Conference of Information Technology, Computer Engineering and Management Sciences, pp. 245–249 (2011)
11. White, S.C.: Decision-support systems in dentistry. J. Dent. Educ. **60**(1), 47–63 (2012)

12. Brickley, M.R., Shepherd, J.P., Armstrong, R.A.: Neural networks: a new technique for development of decision support systems in dentistry. J. Dent. **26**(4), 305–309 (2013)
13. Suebnukarn, S., Rungcharoenporn, N., Sangsuratham, S.: A Bayesian decision support model for assessment of endodontic treatment outcome. Oral Surg. Oral Med. Oral Pathol. Oral Radiol. Endod. **106**(3), e48–e58 (2014)
14. Bai, Y., Zhuang, H., Roth, Z.S.: Fuzzy logic control to suppress noises and coupling effects in a laser tracking system. IEEE Trans. Control Syst. Technol. **13**(1), 113–121 (2005)
15. Precup, R.-E., David, R.-C., Petriu, M.E., Radac, M.-B., Preitl, S.: Fuzzy control systems with reduced parametric sensitivity based on simulated annealing. IEEE Trans. Ind. Electron. **59**(2), 3049–3061 (2012)
16. Antonelli, G., Chiaverini, S., Fusco, G.: A fuzzy-logic-based approach for mobile robot path tracking. IEEE Trans. Fuzzy Syst. **15**(2), 211–221 (2007)
17. Qin, L., Wang, J., Li, H., Sun, Y., Li, S.: An approach to improve the performance of simulated annealing algorithm utilizing the variable universe adaptive fuzzy logic system. IEEE Access **5**, 18155–18165 (2017). ISSN 2169-3536
18. Magoa, V.K., Bhatia, N., Bhatiac, A., Mago, A.: Clinical decision support system for dental treatment. J. Comput. Sci. **3**, 254–261 (2012)

Advanced Machine Learning and Applications

Design and Implementation of IoT Platform for Real Time Systems

M. Hussein[1](✉), M. Zorkany[1], and Neamat S. Abdel Kader[2]

[1] National Telecommunication Institute (NTI), Nasr City, Egypt
mah.hussein@nti.sci.eg
[2] Faculty of Engineering, Cairo University, Giza, Egypt
nemat2000@hotmail.com

Abstract. The internet of things (IoT) is the challenging key to making any system to be smart. Real-time operating systems (RTOS) are used for meeting the real-time requirements for the real-time systems; this makes the IoT applications have the RTOS features like reliability. There are many IoT Platforms have been created, but the most of them are done for specific applications and don't handle the real-time constraints of the real-time systems. In this paper, a design and implementation for an IoT Platform will be proposed for general applications, especially the critical applications for the real-time systems. Also, Many Communication Protocols are used in the IoT Platforms as the Message Queue Telemetry Transport (MQTT) protocol, but many of them don't support the Multi-Topic Messaging. In this paper a simple communication protocol will be presented, it enhances the delay and the traffic needed for the multi-topic messaging by supporting the multi-topic feature, a performance analysis for the proposed protocol versus the MQTT will be studied.

1 Introduction

Internet of Things (IoT) is widely used as a survive technology and its makes things interconnected utilizing the internet as a backbone. IoT used in many fields including E-Health, Smart Metering, Security and Emergencies, Logistics, Industrial Control, Retail, Agriculture, Farming, Home Automation, and Smart Cities [1]. Many IoT Systems are real-time systems (RTSs) which depend on Real Time Operating Systems (RTOSs). RTOS is an Operating System for embedded real-time systems; it is used for its essential features such as; reliability, modularity, predictability, compact, and high performance [2]. RTOS provides multi-tasking and introduces many services; task management, memory management, and time management. For all mentioned features for the RTOS, it should be used in the IoT Platforms. The researchers that use the RTOS formed many research directions, such as; in IoT implementations, IoT frameworks proposing, IoT protocols performance evaluation, IoT RTOS Picking, and RTOS Adapting to work with IoT [3]. Some of IoT researchers are targeting a platform for the IoT, but most of them aren't suitable for real-time systems where time is critical, and it were done for specific applications, its used a ready-made communication protocol such as the MQTT protocol. In this paper, a design and implementation for an IoT Platform will be proposed for general applications and the critical applications, its

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 171–180, 2018.
https://doi.org/10.1007/978-3-319-74690-6_17

implemented for the real-time systems. A communication protocol is introduced for the IoT. It supports the multi-topic messaging, and then it enhances the delay and the traffic needed for the multi-topic messages, a performance evaluation for it and the most common protocols (MQTT) is shown. The rest of this paper is organized as follow. Section 2 presents the related work to IoT platforms. Section 3 discusses Message Queue Telemetry Transport (MQTT) briefly. A design and implementation of an IoT Platform is presented in Sect. 4. Section 5 presents the experiment setup for the hardware/the software parts. The experimental results are presented in Sect. 6.

2 Related Work

The IoT device has the purpose of connecting to other IoT devices for information exchange. The IoT platforms are connecting the device sensors to the data networks, and it enables the developers to make and deploy IoT. Many types of research were done to serve for the IoT applications considering the IoT platforms and other publications focused on the comparison between the IoT different communication protocols like MQTT and other protocols, in some, there are no presented IoT platforms, and the others used a ready-made IoT Platforms. The authors in [4], performed a performance evaluation for four popular pub/sub protocols: AMQP, MQTT, XMPP, and ZeroMQ, they did the comparison using trace files from sensor readings without IoT platforms, they studied for the throughput and delayed parameters. In [5], the MQTT and CoAP performance were studied, and the test parameters are the end-to-end delay and the bandwidth consumption, a common middleware software was implemented, a ready-made board (Beagle Board) was used. A lot of IoT platforms were implemented by authors for specific applications, and they used the ready MQTT communication protocol and the ready-made embedded hardware kits, for example, the authors do the healthcare and medical applications in [6, 7], they didn't use an RTOS in their ready-made platforms. A generic platform is presented using mobile processor and a non-real time operating system in [8]. In this paper, a design and implementation for an IoT Platform are done to solve the rising issues on the other platforms.

3 Message Queue Telemetry Transport (MQTT)

An IoT device board is implemented and worked with the proposed communication protocol which is similar to MQTT, and the performance analysis is done between these two protocols. There are many protocols used to implement the IoT like; CoAP, MQTT, AMQP, and XMPP [9]. MQTT protocol is OASIS Standard; it is a designed protocol for embedded constrained devices which has limited resources and capabilities, it works with the publish/subscribe architecture as shown in Fig. 1. It designed for reliable messaging. The client/Server architecture is the used architecture for the MQTT. The pub/subsystems are working on topics, the subscriber to a specific topic is saved in the broker as an interested in the message on the same topic when a message published on that topic. It will be published directly back to the subscriber; the message is decoupled for the other subscribers for that topic. The MQTT has different message

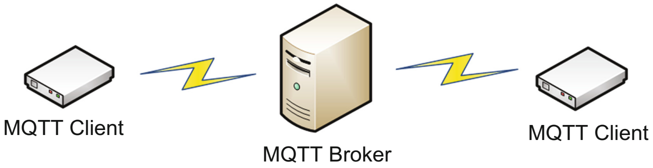


Fig. 1. MQTT architecure.

delivery guarantee methods called quality of service (QoS). The MQTT protocol has light overhead, and extra processing is needed to make the packets less in bytes, it worked perfectly for message delivering techniques. A message with one topic is called single-topic message, MQTT supports single-topic messages but doesn't support multi-topic messages which have more than one topic in the same message [10].

4 Proposed IoT Platform Implementation

Figure 2 shows that the applications built using the IoT Platform have the main RTOS services, the hardware layer is related to sensors, actuators, and the communication modules. To build the IoT System, it should has the following subsystems; the prototyping IoT Node Platform which enables the IoT development, the IoT Server which communicates between nodes, the frame structure which clarifies the how data is exchanged, and the communication protocol which manages the messages transfer between the IoT Nodes and the Server as shown in Fig. 3.

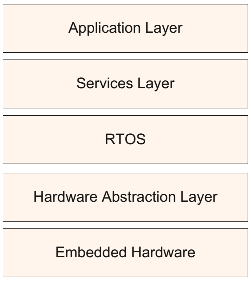


Fig. 2. IoT platform layering architecture.

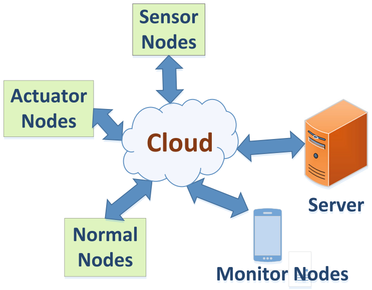


Fig. 3. The proposed IoT system architecture.

A wide range of smart IoT applications can efficiently use the presented system; it provides reliability and IoT connectivity, it uses the WIFI module for TCP/IP Stack which makes the hardware cost to be cheap. The proposed IoT platform is implemented using the ARM Cortex-M4 which is robust and suitable power consumption; the proposed IoT node could be a smartphone. The IoT nodes and the main server are the basic components of the proposed system. The Main Server is responsible for the communication between the system nodes using the Internet as a backbone; the proposed system nodes are classified into main four types. (1) Sensor nodes which sense the environment, (2) Actuator nodes which affect the environment, (3) Normal or Hybrid nodes which sense and affect the environment, and (4) Monitor nodes. The proposed system is discussed in more details in the following subsections.

4.1 Proposed IoT Nodes

The Proposed Platform is an embedded system which could be designed for any type of controllers which meet your system requirements, it was designed and implemented in the PCB Laboratory and passed the expirements successfully, the proposed prototype is implemented using STM Nucleo Board, which uses the ARM Cortex M4 processor, as shown in Fig. 4, the ARM processor is developed especially for high-performance, low power consumption and low cost devices, so it is suitable for the IoT Nodes. Each node consists merely of different units like; a controller which is needed to manage the node tasks, WIFI hardware stack which is used to connect to the wireless network for internet access, sensors to sense the environment, and actuators to affect the environment, as shown in Fig. 5. All system nodes get connected to the main server to do specific tasks so that they could be categorized into four main types; Sensor Nodes, Actuator Nodes, Normal Nodes, and Monitor Nodes. The Sensor nodes sense the environment and transmit the sensed data to the server periodically with a specific configuration period, they are the nodes which include a sensor or more and don’t include any actuators. Actuator nodes affect the environment based on a command from the monitor nodes; they are the nodes which contain an actuator or more and don’t include any sensors. Normal nodes have the sensor and the actuator nodes functionalities. They are the nodes which include sensors and actuators, as shown in Fig. 6, a node behavior is to communicate first with the IoT Server and then sends its sensor data to the server and receives and executes the commands from the monitor nodes through the server. The monitor nodes, as shown in Fig. 3, could be smartphones which control and monitor the system nodes. They are the nodes which don’t include sensors and actuators, but they control the actuators and monitor the sensors readings by sending commands, and receiving and processing the sensor data, as shown in Fig. 7.

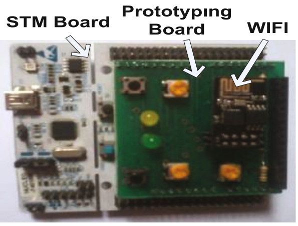


Fig. 4. The proposed IoT node

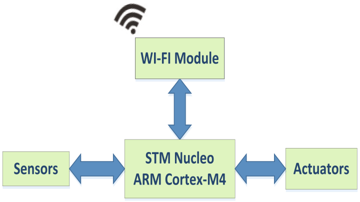


Fig. 5. The IoT node architecture.

4.2 IoT Server

The IoT Server is the main block in the system which communicates all the IoT Nodes together in the IoT system, the server is able to communicate with all types of nodes and enables monitor nodes to visualize sensor data. if the communicated node is a sensor node, the server will forward its data to the registered monitors, else if the node is actuator node, the server will sends the monitor commands to it, else if the node is normal node, the server will do the both issues for the sensor and the actuator nodes, else if the node is monitor node, the server receives the commands and forwards to the actuator nodes, and forwards the sensor data to this monitor node, as shown in Fig. 8.

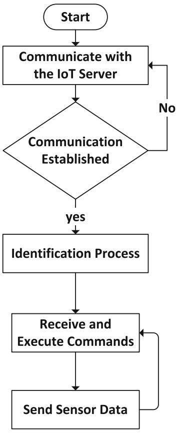

Start
Communicate with the IoT Server
Communication Established
No
yes
Identification Process
Receive and Execute Commands
Send Sensor Data

Fig. 6. Normal node flow chart.

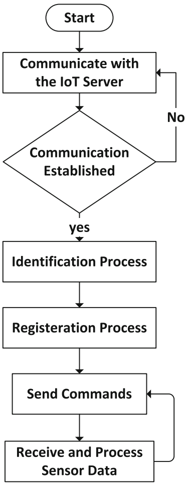

Start
Communicate with the IoT Server
Communication Established
No
yes
Identification Process
Registeration Process
Send Commands
Receive and Process Sensor Data

Fig. 7. Monitor node flow chart.

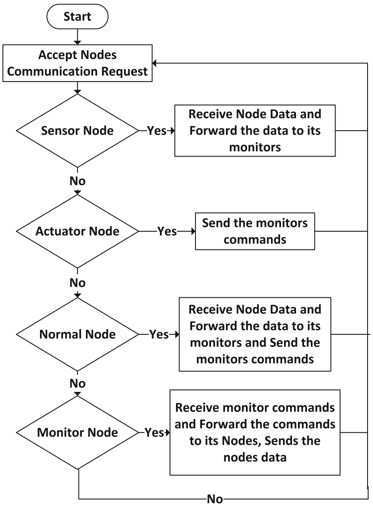

Start
Accept Nodes Communication Request
Sensor Node
Yes
Receive Node Data and Forward the data to its monitors
No
Actuator Node
Yes
Send the monitors commands
No
Normal Node
Yes
Receive Node Data and Forward the data to its monitors and Send the monitors commands
No
Monitor Node
Yes
Receive monitor commands and Forward the commands to its Nodes, Sends the nodes data
No

Fig. 8. IoT server flow chart.

4.3 Proposed Communication Protocol

The main server is responsible for communicating between system nodes to do the system tasks, all nodes open the TCP connection with the main server and identify itself to the system by sending its identification number to the main server waiting the server acknowledgement (ACK). The sensor node identify itself, then it is ready to send the periodic data or information that it has, the actuator node, after the identification process, receives commands transmitted by the monitor nodes through the main server, the normal Node does the sensor and the actuator nodes functionalities, as shown in Fig. 9. The Monitor Node, after the identification process, it registers for receiving data from a specific sensor nodes and sends its commands to certain actuator nodes, as shown in Fig. 10. Node to Node communication is done through the server, each node tries to connect with the server and identify itself, after receiving the server ACK, they can successfully communicate with each others, as shown in Fig. 11.

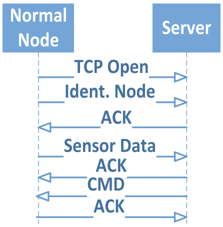


Fig. 9. Normal node sequence diagram.

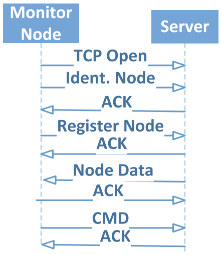


Fig. 10. Monitor node sequence diagram.

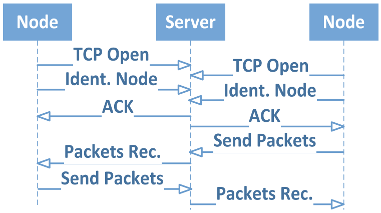


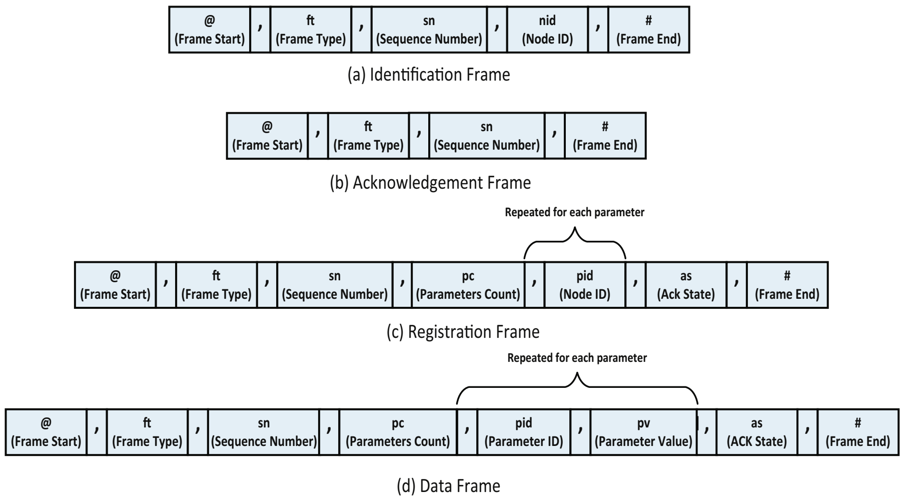
Fig. 11. Node to node communication.

4.4 Frame Format

The main communication method of the nodes is the WIFI technology, which is implemented by using WIFI hardware module, the module has the built-in TCP/IP Stack. The Proposed IoT System has different frame types including identification frame, ACK frame, registration frame, and data frame, (refer to Fig. 12). Table 1 shows the standard frame fields and the frame types used in the proposed frame structure. Identification frame as shown in Fig. 12(a) shows the frame structure needed for node identification and the node should transmit this frame at first to declare itself to the server with a specific identification number (ID) and should receive an ACK frame as a response from the server, a node can receive or transmit frames after that. While registration frame works when a node needs to listen to certain parameter. It should transmit the registration frame with PC and PIDs as shown in Fig. 12(b), after the registration process, any data sent by other nodes for that parameters, it will be sent to the node by the server, the ID Frame should be sent at first. Acknowledgement Frame as shown in Fig. 12(c) shows the Acknowledgement (ACK) frame which is sent back to the sender to acknowledge the transmission process. The ACK frame received if the requested ACK field is set. An ID frame must be followed by ACK frame from the server side. In data frame, a node can share its data to other nodes by transmitting the data frame. As shown in Fig. 12(d), it will specify multiple data parameters with parameter count field (PC). Each parameter has an id (PID) and value (PV), the other nodes want to receive the data, it should register for these parameters using PID, the ID Frame should be sent at first.

Table 1. Common frame fields description and frame types

Common frame fields description		Frame types	
Frame field	Field description	Type	Description
@	Start of frame	I	Identification frame
ft	Frame type	R	Registration frame
nid	Node ID	A	Acknowledgement frame
#	End of frame	D	Data frame



@ (Frame Start)	,	ft (Frame Type)	,	sn (Sequence Number)	,	nid (Node ID)	,	# (Frame End)

(a) Identification Frame

@ (Frame Start)	,	ft (Frame Type)	,	sn (Sequence Number)	,	# (Frame End)

(b) Acknowledgement Frame

@ (Frame Start)	,	ft (Frame Type)	,	sn (Sequence Number)	,	pc (Parameters Count)	,	pid (Node ID)	,	as (Ack State)	,	# (Frame End)

(c) Registration Frame

@ (Frame Start)	,	ft (Frame Type)	,	sn (Sequence Number)	,	pc (Parameters Count)	,	pid (Parameter ID)	,	pv (Parameter Value)	,	as (ACK State)	,	# (Frame End)

(d) Data Frame

Fig. 12. System frames architecture.

5 Experiment Setup

Experiments are done to study the proposed communication protocol performance and the MQTT protocol performance by setting different network parameters which affect the performance of the protocol; the experiments are done in a real working network infrastructure of the NTI. The performance metrics measured are the delay and the total transmitted bytes per successfully transmitted message, the delay is defined as the time interval between the publishing of a message and the returned ACK from the server.

The setup, as shown in Fig. 13, has three machines are used, a laptop which executes the WANem software to affects the network for applying channel losses and communication delays, a PC acts as the server to enable the platforms for communicating each other, and another PC which acts as a Node to publish the message waiting its acknowledge. The PC which acts as an anode has the Wireshark Software running to catch the traffic for later analysis. The two protocols are running on the server machine, every message is published by the Node will go to the server through the WANem machine, and the server's ACK is returned to the Node through the WANem machine too, as shown in Fig. 14.

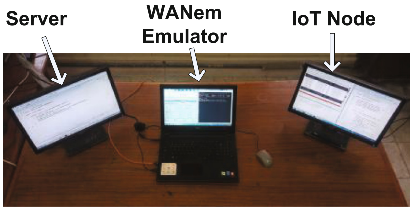


Fig. 13. The experiment hardware setup.

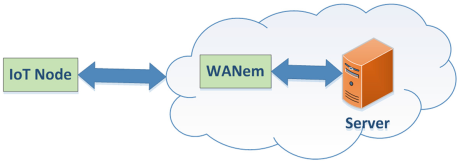


Fig. 14. The data flow.

The mosquitto open source implementation is used for the MQTT protocol, it will be tested on the server as will as the proposed communication protocol. For accurate delay calculations and packet length measurements, the wireshark software will be used.

6 Experimental Results

For the simple setup of the experiment which is one node and one server, the both protocols achieve the delivering of each message without caring about the packet loss percentage applied, this means that the both protocols has a good message delivery technique to work with the different packet loss rates, so the performance measurements will be studied for the message delay and the total amount of the data transmitted for each successfully transmitted message. The message delay is an important metric specially for the real time systems, where the time is more critical, the applied packet loss rates will have an impact on the message delay due to the message re-transmission for successfully delivering of the message, MQTT protocol with quality of service 1 is compared with the proposed communication protocol with ACK state 1 for the same messages. MQTT has lower packet size than the proposed protocol, so its message delay is lower than the proposed protocol, as shown in Table 2, and with a network losses, the proposed protocol is still higher delay than the MQTT protocol, as shown in Fig. 15.

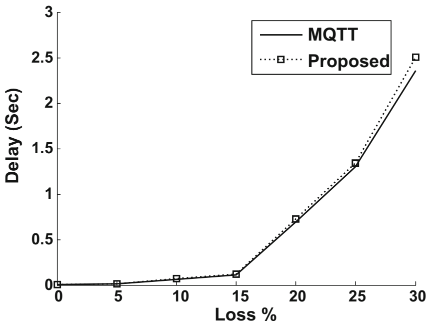


Fig. 15. The average message delay (Sec).

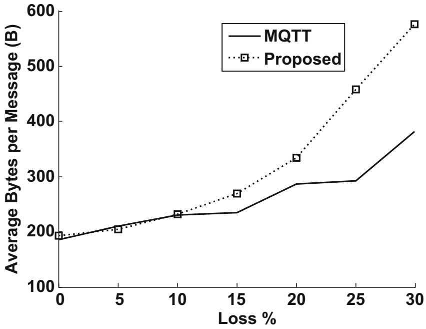


Fig. 16. Average total traffic per message (Bytes).

The network traffic generated should be small as possible, it means that the message data transferred per message is also an important metric, which is calculated as the total bytes generated divided by the number of the successfully delivered messages, it is calculated using Wireshark for different packet loss percentages. As MQTT protocol has lower packet overhead, its message size is lower than the proposed protocol, in network losses, for multiple re-transmissions, the MQTT has lower bytes needed than the proposed, as shown in Fig. 16. In the previous subsections, the single-topic

Table 2. Average end-to-end delay (Sec) for different losses (%).

	0% Loss	10% Loss	20% Loss	30% Loss
MQTT	0.001081	0.065235	0.695212	2.357077
Proposed	0.001095	0.071645	0.721798	2.505362

messages were studied, in some cases, may a node needs to publish different topics to different nodes at the same time. in such case, the MQTT protocol doesn't support the multi-topic messages, so it publishes each topic within a new message, but the proposed protocol supports the multi-topic messages, so its publishes multiple topics to different nodes in a single message. The Multi-topic Message Delay is shown in Fig. 17, and the Bytes per Multi-topic Message is shown in Fig. 18 for no losses. From Figs. 17 and 18, the proposed protocol is lower delay and lower traffic than the MQTT protocol due to the multi-topic feature which adds lower bytes for more added topics than the MQTT.

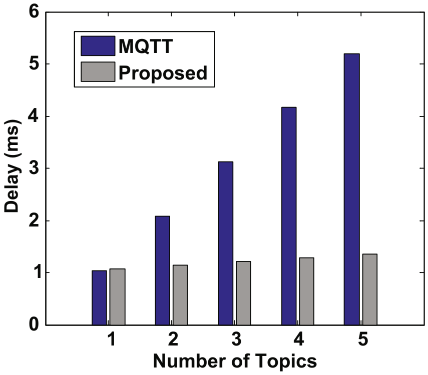


Fig. 17. The multi-topic message delay (ms).

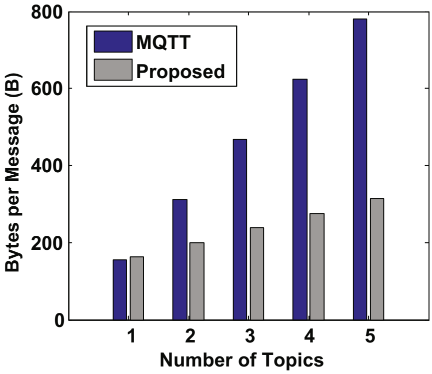


Fig. 18. Total traffic per multi-topic message (Bytes).

7 Conclusion

In this paper, an IoT Platform was presented using RTOS, and a communication protocol was proposed, the proposed platform and the proposed communication protocol were tested on a practical sceinario. A performance analysis is done for the proposed communication protocol and the MQTT protocol. The proposed protocol supports the multi-topic message but the MQTT not, this makes MQTT to publish multiple of messages equal to the number of topics, but the proposed protocol publishes one multi-topic message, so the proposed protocol is suitable for the multi-topic messages. The proposed IoT platform is suitable for many applications especially the critical applications due to the RTOS used, for the multi-topic messages, the proposed protocol is lower delay and lower traffic than the MQTT protocol due to the multi-topic feature which adds lower bytes for more added topics than the MQTT.

References

1. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of Things (IoT): a vision, architectural elements, and future directions. Future Generat.Comput. Syst. **29**(7), 1645–1660 (2013)
2. Labrosse, J.: MicoC/OS-II The Real-Time Kernel, 2nd edn. Taylor & Francis, London (2002)
3. Hussein, M., Zorkany, M., Abdelkader, N.: Real time operating systems for the internet of things; framework. In: IEEE World Symposium on Computer Applications & Research (WSCAR), Egypt (2016)
4. Happ, D., Karowski, N., Menzel, T., Handziski, V., Wolisz, A.: Meeting IoT platform requirements with open pub/sub solutions. In: Annals of Telecommunications. Springer (2017)
5. Thangavel, D., Ma, X., Valera, A., Tan, H., Tan, C.: Performance evaluation of MQTT and CoAP via a common middleware. In: IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 21–24 April 2014, Singapore (2014)
6. Zamfir, M., Florian, V., Stanciu, A., Neagu, G., Preda, Ş., Militaru, G.: Towards a platform for prototyping IoT health monitoring services. In: Borangiu, T., Dragoicea, M., Nóvoa, H. (eds.) Exploring Services Science, IESS 2016. Lecture Notes in Business Information Processing, vol. 247. Springer, Cham (2016)
7. Jamin, A., Fasquel, J., Lhommeau, M., Cornet, E., Lacourtoisie, S., Henni, S., Leftheriotis, G.: An aggregation plateform for IoT-based healthcare: illustration for bioimpedancemetry, temperature and fatigue level monitoring. In: International Conference on IoT Technologies for HealthCare. Springer (2016)
8. Park, K., Kim, S.: Design of mainframe for IoT framework. In: International Conference on u- and e-Service, Science and Technology (2015)
9. Karagiannis, V., Chatzimisios, P., Vazquez, F., Alonso, J.: A survey on application layer protocols for the internet of things. Trans. IoT Cloud Comput. **1**(1), 11–17 (2015)
10. IBM, MQTT V3.1 Protocol Specification, Protocol Standard

Machine Learning: A Convergence of Emerging Technologies in Computing

Morteza Kiadi[1] and Qing Tan[2](✉)

[1] Faculty of Continuing Education and Training, Seneca College, Toronto, Canada
morteza.kiadi@senecacollege.ca

[2] School of Computing and Information Systems, Athabasca University, Athabasca, Canada
qingt@athabascau.ca

Abstract. Machine Learning (ML) is the convergence of different disciplines in science and technology. While it is conceived, ML is part of computer science however, in its essence, it borrows or utilizes methods from other classic disciplines and mature computing theories and technologies, such as statistics, computational algorithms, optimization, and data mining. In this paper, we explore how these disciplines and technologies work hand in hand to prepare a passionate researcher gains a comprehensive perspective for being an ML expert. We have proposed a roadmap to show how different disciplines and technologies contribute to the ML foundation and we discuss each part of the roadmap separately. Moreover, to apply the proposed roadmap in practical terms, we also present how to use the proposed roadmap in the context of IoT and Fog Computing. The main contribution of this paper is to provide a guideline by developing a roadmap for foundational requirements of being a Machine Learning subject matter expert for the researchers or industry experts.

Keywords: Machine Learning · Deep Learning · Fog Computing · IoT
Algorithms · Optimization · Statistics

1 Introduction

In this paper, we explore fundamental requirements of being a Machine Learning subject matter expert. We introduce how different disciplines and technologies play roles in this newly emerging computing technology. We show Machine Learning as a multifaceted technique and knowledge to be applied to different domains. Being expert in ML needs multi-year dedication to get familiar with advanced topics in computer science such as algorithms and optimization, artificial intelligence, data mining as well as mathematics and statistics. After all, these fundamental knowledges should be applied to different domains such as security, bioinformatics, smart city, Cloud Computing and more. In this paper, we categorize these activities into two domains (1) core Machine Learning and (2) business acumen (the domain that ML techniques will be applied to).

While Machine Learning is considered a collective know-how that utilizes computer science skills and statistics/mathematical foundations, a data scientist has to have ML

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 181–192, 2018.
https://doi.org/10.1007/978-3-319-74690-6_18

skills plus business acumen at the same time. Figure 1 shows how data scientists and ML are related together.

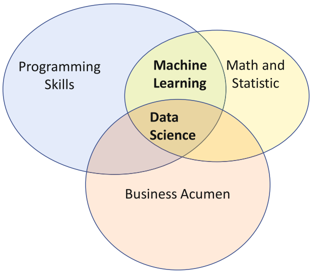


Fig. 1. Being a data scientist is more than knowing ML techniques inspired by [1]

In this article, we present how to apply ML techniques to a business acumen formulated in the form of IoT and Fog Computing. For this reason, before diving into the main topic of this article (Machine Learning), in Sect. 2, a brief background about a business domain (IoT and Fog Computing) has been provided. In Sect. 3, we present how ML can be applied to Fog Computing. In Sect. 4 we discuss why optimization is important in Fog Computing. In Sect. 5 and its sub sections we introduce a roadmap to toward ML proficiency. Finally, in the last section of this paper, we conclude with a series of suggestions and new research areas to apply ML in different areas of Fog Computing.

2 IoT and Fog Computing

The classical Cloud Computing is designed and architected around a few assumptions that they are not necessarily applicable to the IoT paradigm because of its centralized computing services. With the invention of smart phone, more recently the adoption of wearable devices, and the proliferation of IoT devices, the Cloud Computing paradigms have to be changed, which leads the birth of fog computing.

The IoT applications recently have been extended to different areas and industries. To name a few, IoT devices and Fog Computing can be found in autonomous vehicle, smart factory, smart home, smart city, smart grids and connected farms, smart traffic light systems, drone, augmented reality and healthcare. In general, the IoT devices have been attributed with the following items:

Form Factor: The IOT devices, sometimes known as the Uniquely Addressable Heterogeneous Electronics (UAHE) devices, generate and consume data in different shapes and forms. To name a few, they may be in the form of location-aware mobile vehicles that are connected to traffic systems; or in the form of smart devices geographically distributed and connected to react to specific event or condition in specific area; or in the form of human wearable devices to sense and collect human body signals for real-time monitoring, responding or further analysis.

Network Bandwidth: Furthermore, IoT devices usually do not have high speed bandwidth to connect to the Cloud, while most of the times, they require real-time response.

Time-Constraint: IoT devices generate many small size data in short periods of time and sensitive to their processing time.

Resource constraint: Since IoT devices are scarce in processing power, memory, and battery life, they may need to offload/outsource part of their tasks to another device in the Cloud.

Mobility: Since IoT devices can be movable, the location data plays a big role in the analytics.

Heterogeneity: IoT devices are in different forms. They may be in the form of sensors, mobile devices, wearables, actuators or edge devices connected to Fog access layer.

All these attributes call for rethinking about the computational delivery models in IoT. To make sure IoT devices interact with each other or reacting to events in a timely manner, researchers and industry leaders have suggested a different computational delivery model that it is physically closer to IoT devices.

Having a computational center that is "near" to IoT devices, contributes to faster response time and less latency. The idea is to have data centers that they are smaller in size and more distributed geographically while they are in proximity of the IoT devices. This is where the term Fog Computing comes to play. Fog data centers are like conductors to the Cloud data centers. They either fulfil IoT requests internally or they pass them through to the Cloud data centers.

3 Machine Learning (Core) and Fog Computing (Business Acumen)

Recent developments in Cloud Computing are the result of innovating new or optimizing existing algorithms to build scalable infrastructure in traditional computing data centers to handle different workloads in a very cost-effective manner. The application of algorithms varies, and they serve Cloud Computing in different aspects such as cost optimization, scheduling, demand analysis, intelligent bandwidth management, machine learning, monitoring, capacity planning, security, etc. IoT devices usually need to be small in their sizes and fast in processing and response time, i.e. real-time. ML techniques can be used to optimize the algorithms for Fog data center to meet such needs. The first step in applying ML techniques to Fog Computing is to identify the questions that we are trying to answer by ML. The below list is a candidate list we have developed with this objective:

(A) What algorithms and techniques play role in resource scheduling and optimization in Cloud Computing.

(B) Which of the identified algorithms and techniques in step A are related or could be related to the selected area of research, i.e. Machine learning in Fog Computing.

(C) Which of the algorithm optimization and approximation techniques (such as computational intelligence, evolutionary computation or swarm intelligence) are qualified in building scalable and cost-effective Fog systems.
(D) Short list one or two algorithms in step C as candidates for optimization in the context of the ML. The optimization can be considered from time, processing speed; resource consumption (CPU or memory); or IOPS.
(E) Is there a better way to develop new algorithm(s) to perform the same tasks? The "better" in this context can be translated to faster or less resource intensive.

4 Optimization in Fog Computing for IoT

Since Fog data centers are not as resourceful as Cloud data centers and they should response in real-time, the "optimization" has a significant role in Fog Computing. There are different types of optimization algorithms that they are known in mathematics for years such as Gradient Decent and Newton's methods, suitable for more complex problems; or unconventional optimization algorithms [2], such as linear programming, Quadratic Programming, Combinatorial Optimization, and Nonlinear Programming.

The class of algorithms that they work on optimization techniques are used in different applications such as distributed meeting scheduling, distributed combinatorial auctions, overlay network optimization, distributed resource allocation and airports takeoff and landing slot allocation. Optimization problem in allocating resources to IoT devices for real-time processing are very compatible to the above-mentioned optimization problems and we believe those techniques should be reviewed in the context of IoT and Fog Computing.

5 Roadmap to Machine Learning Proficiency

The following roadmap is suggested with the objective of preparing an individual as a ML expert. The roadmap suggests a few practical topics to be studied initially in this journey. On the other hand, while the researcher study on the IoT and Fog Computing (or any business domain that the researcher is interested in), it is expected to narrow down this roadmap to have it focused on an interesting and important topic. The goal is to find how to apply ML algorithms to Fog Computing for IoT. Figure 2 shows this high-level roadmap.

5.1 Fundamental Theories

The first step is to gain the necessary knowledge about the fundamental theories around Machine Learning. There are many resources that claim to teach ML without getting into mathematical and statistical theories. They have done a good job by not to deep dive into those topics. However, there is no Machine Learning textbook that does not have chapters in mathematical and statistical topics as well as basic data structures and algorithms. This task needs to be extended by stepping into the advanced topics in algorithms and specifically in NP-hard problems. After that, it would be useful to learn about

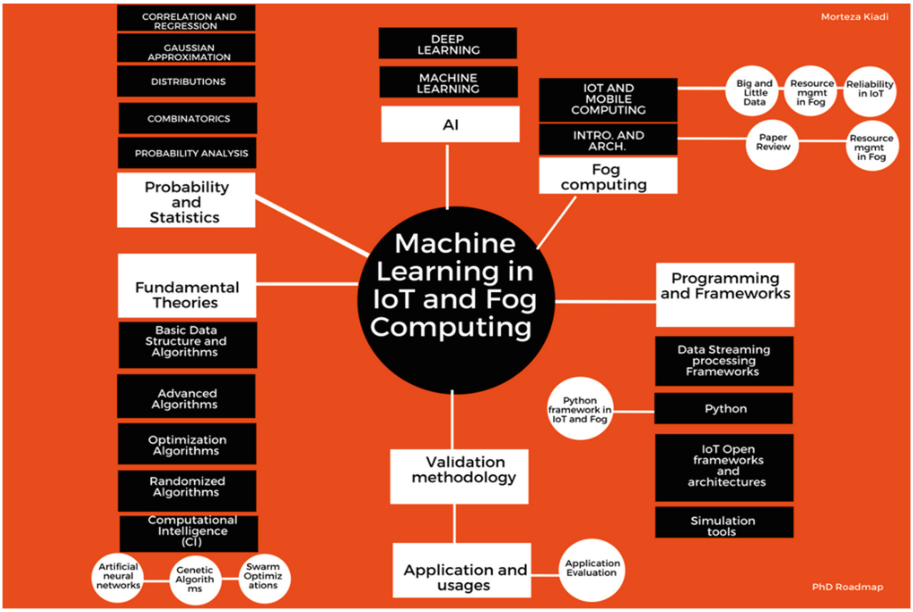


Fig. 2. A proposed roadmap to ML proficiency

optimization algorithms and techniques. As it is discussed in Sect. 4, the reason we believe optimization algorithms play an important role in Fog Computing for IoT is due to the essence of time-sensitivity or time constraint of IoT traffics. The sensors continuously generate many small pieces of data that they must be processed in a short period of time. Such constraints usually enforce us to optimize the way we make decision and we communicate. For example, many IoT devices like wearable or Fitbit naturally move from one location to another. Such devices generate, store and send data to processing centers, data centers or edge locations, and wait for response. But processing and decision making can be complex and sometimes it needs coordination with other agents in the whole ecosystem and as such, we cannot just decide autonomously without considering other moving parts in the system. For that reason, we cannot even assume we can collect every agent's information in a centralized repository since they do not have a fixed location. On the other hand, the decentralised agents need to communicate and share knowledge with other agents to make an optimized decision. But communication among all agents to provide the same/cohesive view of the system's state will be very time consuming and it will not fit with the temporal constrained of IoT events and response requirements. Furthermore, we should consider other constrains such as "resource constrain" (if the required resource to process the IoT event is available) or "order constraint" (if the events should be processed in order). Such limitations impose considering optimization algorithms and its methods. That is the reason we should have included that in the skillset. In general, optimization can be defined as finding solution for a problem where it is necessary to maximize or minimize a single or set of objective functions within a domain with acceptable values and with some restrictions. There are different types of optimization algorithms that we have shown a summarized taxonomy in the Fig. 3. A more comprehensive version of the optimization algorithms is illustrated in the [3].

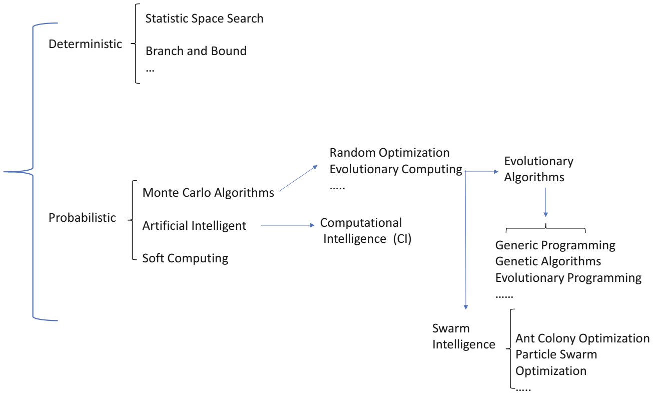


Fig. 3. Optimization algorithm taxonomy inspired based on [3]

As you see in the Fig. 3, optimization algorithms are categorized into two main groups: deterministic and probabilistic. When solution space for a problem is clear, we can use deterministic algorithms but when the solution space is not clear, or it is too complex to determine what solutions fit with a problem, we use probabilistic algorithms. Heuristic algorithms in the probabilistic group help us to find the best next step based on the collected data so far. Although we do not suggest any specific probabilistic algorithm, but it seems computational intelligence (CI), Genetic algorithms and swarm optimization are good candidates to start with. As a conclusion we suggest the researchers to find how CI techniques can be used in Cloud Computing and IoT.

5.2 Probability and Statistic

Machine learning at its core tries to find the best mathematical model by evaluating the relationship between the outputs of given inputs. By "best", we mean finding the best function that has minimum error in mapping inputs data to their equivalent outputs. When such optimum model/mathematical function is found, we can use that function "to predict" an unknown output for a new given input. This activity in statistic is called "predictive analysis" and there are well known methods for that for years. Those methods can detect patterns in existing data by uncovering the association between inputs and expected outputs. Examples of scenarios that we need prediction are price prediction, dosage prediction, risk assessment and diagnosis. All these examples use a function to predict and that function is trained to predict as accurate as possible. This is where ML comes to the picture. ML techniques train the prediction function to predict with less error [4]. Thus, Machine Learning is closely related to the fields of statistics and data mining, while it differs slightly in terms of its emphasis and even the terminologies.

Since IoT devices will be producing humongous amount of data in small fractions of times, we need to have fast methods to analyse those data and predict or decide next step. After preparing enough in statistic, the roadmap leads into more advanced topics in machine Learning and specifically in deep learning techniques that it is discussed in step 3.

5.3 Machine Learning and Deep Learning

Machine learning and Deep learning (DL) are two main pillars of data science. Majority of researcher times will be around learning the ML and DL techniques and their associated tools. It appears that learning these two topics need basic knowledge in linear algebra and matrices.

As shown in Fig. 4, Machine Learning is a very broad topic and it includes many techniques. Neural networks, regression, Deep Learning and clustering all among the related topics in Machine Learning. Such broad topic needs a noticeable amount of time to learn and to master. Researchers would need to deep down in this path to learn about different techniques, the application of those techniques and the way they can apply those techniques in addressing the problems in the Fog computing for IoT.

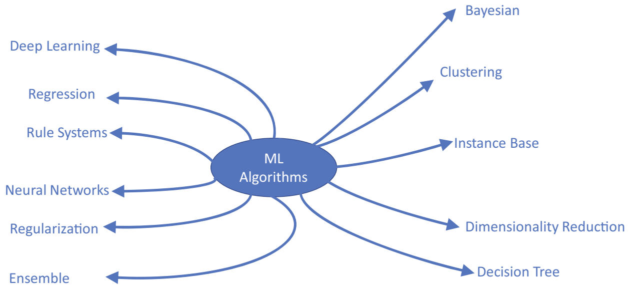


Fig. 4. Machine learning mind map inspired by [5]

As part of preparation of this article we had selected a few of most practical techniques in ML and we introduce them briefly in the following paragraphs:

- Gradient Descent: an optimization algorithm that uses derivatives to locate local minimum.
- Linear Regression: a parametric machine learning algorithm that its root is in statistic and it is used to find the relation between input and output numerical values. This algorithm is used to find the prediction function.
- Logistic Regression: logistic regression is also borrowed from statistic. It maps the data to a binary value and it is used for binary classification.
- Linear Discriminant Analysis: a parametric machine learning algorithm. While the logistic regression is a classification algorithm that is limited to only two-class classification problems, the Linear Discriminant Analysis is the preferred linear classification when we have more than two classes.

- Classification and Regression Trees (CART): a nonparametric machine learning algorithm to learn from data and make a tree model to be used in predicting values. CART is the base algorithm for other ML algorithms like bagged decision trees, Random Forests and Boosted decision trees.
- Naïve Bayes: a nonparametric machine learning algorithm that again its root is based on a statistic formula called Bayes' theorem. In this model, we would like to find the prediction model given specific conditions. Naïve Bayes is a classifier algorithm for binary and multiclass problems.
- K-Nearest Neighbors: one of the important difference between KNN and the other algorithms is that in KNN we do not have training data set and test data set separately. We build a data structure based on entire data and then we use it for classification and prediction of new data. When a new data is given, its distance to similar data points or its neighbours is calculated, and it will be classified to the closest neighbour.
- Support Vector Machines (SVM): a nonparametric machine learning algorithm that is used for classification. The optimized SVM model should be found through numerical optimization methods but since that is not efficient usually other algorithms like Gradient decent comes to rescue to find optimized coefficients for the SVM prediction function.
- Boosting: a method to create a strong classifier from a few weak classifiers. In this model, the second model improves the first model and third model improves the second models and son on.
- Deep Learning: Deep Learning is one of the algorithms in ML that it learns from training data by utilizing specific techniques/algorithms in Artificial Neural Networks (ANN). Deep Learning allows us to represent/model the data in many layers, hence the word "Deep" in Deep Learning, and predict the future outcomes by learning from existing data, i.e. training set. Figure 5 shows a two-layer ANN. A good model predicts with less or zero error while the objective is "to find" such models with its corresponding parameters. In Deep Learning, these layered representations are modeled via ANNs. As a result, ANN is a method to implement Deep Learning. ANN by far is characterized as one of the "computational models" that can learn from complex data to generalize, classify or cluster them. Mathematically speaking, learning is about "searching" and "selecting" the best parameters, P1, P2, Pn in Fig. 5, to adjust the computational model/prediction function in the solution space and to minimize the errors in next predictions. After selecting the best candidate

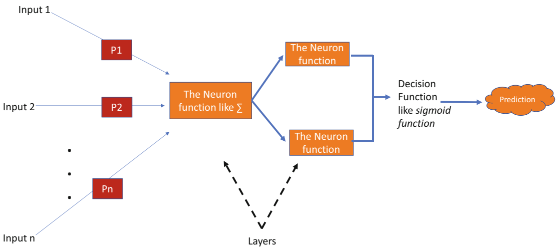


Fig. 5. A neural network with 2 layers

parameters, ANN is recurred (executed again) to predict and then to measure the errors one more time. This process is continued until it predicts more precisely with less error.

Recently Deep Learning was center of attention for many data scientist activities due to its flexibility in modeling complex models in real world problems.

5.4 Software Tools and Programming Languages

Thus far, there are lots of attention around Python language in the field of machine learning and there are ample resources to speed up practically.

In [6] the "Plethora of Tools for Machine Learning" there is a good compilation of tools and libraries that are needed for machine learning. Based on our research, the TensorFlow and Theano and Keras are three main Python libraries used in Machine learning. In addition, it seems MATLAB and R are good alternatives to programming languages.

In the "Overview of Machine Learning Tools" [7] there is list of available tools and libraries around machine learning. In that list, R is number one followed by Microsoft Excel. In addition to machine learning tools, simulation tools are needed to simulate and generate data in the selected application domain. For example, in our hypothetical example, we need to simulate the IoT and smart device traffics in Fog Computing data centers through simulators. We came across the CloudSim toolkit [8] for simulation of data center traffics.

5.5 Fog Computing and IoT

Although Machine leaning is one of the hot topics in recent years, it would not be valuable if you do not apply that in practical terms and in real problems. As we discussed before in this paper, we apply ML techniques to IoT and Fog processing centers and we try to find problems that machine learning and deep learning have answer/solutions in that domain.

The Fog Computing terms is coined by Cisco in 2011 to extend Cloud Computing to edge network. After that, many groups around the globe start extending Fog Computing idea and one of the most active one is the European consortium called OpenFog [9]. Fog Computing is decentralized in their nature and they are more efficient in providing computing and application services to IoT devices due to their proximity to them. As it is shown in Fig. 6, the OpenFog Consortium intends to harmonize with other groups including the Industrial Internet Consortium (IIC), ETSI-MEC (Mobile Edge Computing), Open Connectivity Foundation (OCF) and the OpenNFV.

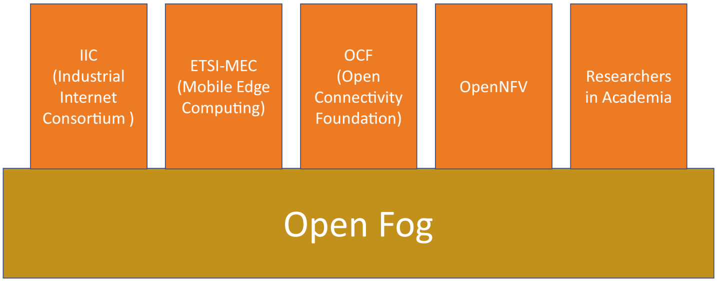

IIC (Industrial Internet Consortium)
ETSI-MEC (Mobile Edge Computing)
OCF (Open Connectivity Foundation)
OpenNFV
Researchers in Academia
Open Fog

Fig. 6. OpenFog and other Consortia inspired by [9]

Fog computing is different from cloud computing in terms of its processing power and proximity to fog devices to fulfil low latency requirements of IoT devices. Carnegie Mellon University in 2013 has coined the word Cloudlets. A cloudlet as the middle-tier in the 3-tier hierarchy of: mobile device→cloudlet→cloud. It is like a "data center in a box" whose goal is to bring cloud closer to the users. Microsoft™ calls them "micro data centers" [10]. The Microsoft approach is to build an extensive infrastructure of micro DCs (1-10s of servers with several TBs of storage, $20K- $200K/mDC) and place them everywhere. If the device cannot find any cloudlet available, then it will send its request to the cloud or, in the worst case, complete the task with its own resources. Thus, a user gets real-time response by low-latency, one-hop, high-bandwidth, and low-cost access to cloudlet. In comparison with cloudlets, we have femtocells that also known as "home base station". Femtocells are deployed indoors to provide good coverage [11]. Regardless which path we choose, we must response back almost in real time that it imposes new constraints to existing algorithms in cloud data centers. [12] compares Cloud and Fog Computing from different viewpoints such as security, latency, mobility and other factors.

5.6 Validation and Application

The idea of applying ML techniques to Cloud and Fog Computing allows us to solve problems that are not easily, if it is not impossible, answerable by explicit programming. We need to expose ourselves to existing problems and the solutions and techniques in Machine and Deep Learning and getting familiar with applications of those solutions in the context of problems. After obtaining such skills, we can utilize and apply the same problem-solving techniques in different fields. Currently Wikipedia [13] has listed more than 30 applications of Machine Learning in different fields such as optimization, robot locomotion, marketing, machine perceptions, computer vision, etc.

6 Conclusion

In this paper, we presented a spectrum of ML techniques and we present how they can be utilized in context of Fog Computing and IoT.

In this paper, we presented how to apply ML to specific business domain. We contributed by illustrating how to utilize ML techniques in IoT and Fog Computing domain. As part of that, we showed it is necessary to define the formal specification of the goals, constraints and satisfactory results in the system under investigation. The results must be supported computationally with enough reasoning automated or theoretically. We emphasized in supporting the logical foundations, by using existing data available publicly on the Internet or to generate data by simulation tools to validate the theories. The testing will validate or compare the models that it is proposed with other existing models or theories.

Here we suggested seven areas in Fog and Cloud Computing that ML algorithms can add value as an example of defining goals in applying ML techniques in Cloud and Fog Computing. The following topics can be researched in the context of Fog Computing and Machine Learning:

- Scheduling and optimized resource management in Fog Computing that is discussed in [14–16]
- Machine learning in Fog Computing
- Computational Intelligence in Cloud and Fog computing
- Communication and SLO between cloudlets
- Optimization algorithms in Fog Computing
- Security issues in Fog Computing
- Optimizing the learning algorithms automation in Fog Computing due to time-constraints

References

1. http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram
2. Brownlee, J.: Clever Algorithms: Nature-Inspired Programming Recipes. Lulu.com, Morrisville (2012)
3. Weise, T.: (2009). http://www.it-weise.de/projects/book.pdf
4. Kelleher, J.D., Namee, B.M., D’Arcy, A.: Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies. MIT Press, Cambridge (2015)
5. Brownlee, J.: (2013). http://machinelearningmastery.com/a-tour-of-machine-learning-algorithms/
6. Steeves, J.: (2015). http://knowm.org/machine-learning-tools-an-overview/
7. Pop, D., Iuhasz, G.: (2013). https://www.ieat.ro/wp-content/uploads/2013/05/technical_reports/IEAT-TR-2011-1.pdf
8. Superwits Academy (2014). http://www.superwits.com/library/cloudsim-simulation-framework
9. Open Fog Consortium. https://www.openfogconsortium.org/resources/#white-papers
10. Bahl, V.: Microsoft Research (2015). https://www.microsoft.com/en-us/research/wp-content/uploads/2016/11/Micro-Data-Centers-mDCs-for-Mobile-Computing-1.pdf
11. De, D.: Mobile Cloud Computing: Architectures, Algorithms and Applications. CRC Press, Boca Raton (2015)
12. Saharan, K.P., Kumar, A.: Fog in comparison to cloud: a survey. Int. J. Comput. Appl. (0975–8887) **122**(3), 10–12 (2015)
13. https://en.wikipedia.org/wiki/Machine_learning

14. Agarwal, S., Yadav, S., Yadav, A.K.: (2016). http://www.mecs-press.org/ijieeb/ijieeb-v8-n1/IJIEEB-V8-N1-6.pdf
15. https://www.researchgate.net/file.PostFileLoader.html?id=57bea7b6217e20e33f730969&assetKey=AS%3A398883168505856%401472112566074
16. Bittencourt, L.F., Diaz-Montes, J., Buyya, R., Rana, O.F., Parashar, M.: (2017). http://www.buyya.com/papers/MAS-Fog2017.pdf

Discrimination of Satellite Signals from Opencast Mining of Mineral Ores of Hematite and Uranium Using Digital Image Processing and Geostatistical Algorithms

Richa N. K. Sharma[1](✉), Roheet Bhatnagar[2], and Abhishek Ojha[1]

[1] Department of Remote Sensing, Birla Institute of Technology, Mesra,
Ranchi 835 215, Jharkhand, India
richasharma.ranchi@gmail.com

[2] Department of Computer Science and Engineering, Manipal University,
Jaipur 303007, Rajasthan, India

Abstract. This study has been conducted on mining areas of Gua and Banduhurang in Jharkhand State, India, Barbil in Orissa State, India and Mines d'Arlit, Niger. Data from the corresponding sensors of EO-I, LANDSAT and IRS (i.e., Hyperion, ETM+ and LISS III respectively) were used to discriminate satellite signals of Hematite and Uranium ores from these locations. For the data of Hyperion being hyper-spectral, correction mechanism were performed through relevant algorithms: Fast Line-of-sight Atmospheric Analysis of Hypercube (FLAASH) to remove atmospheric aerosol effects, Minimum Noise Fraction (MNF) to remove noise, Pixel Purity Index (PPI) to get spectrally the most pure pixel and Spectral Angle Mapper (SAM) in order to match the spectral similarity between an image pixel spectrum and a referenced spectrum. The data achieved from ETM+ had line-stripping, and thus were restored. On the LISS III data, vegetation had to be, virtually, removed from the images of the Indian sites, using Normalized Difference Vegetation Index (NDVI), in order to equate them with that from the Niger site. The processed data was put to a common platform statistically. Segregation of Uranium, a radioactive ore, from Hematite, a non-radioactive iron ore, could be achieved up to 82.35% using TOPSIS and 90% using pair-wise Student's t-Test. The technique of Band Ratio was also carried out and an index was generated to isolate these mines from their surroundings.

Keywords: FLAASH · Pixel · PPI · SAM · NDVI · Ground-truth
TOPSIS · Pair-wise Student's t-Test · Graphic screen · Band
Band ratio · Index image

1 Introduction

Surface mining features related to mineral ores can be discriminated through elements of image interpretation of satellite imagery displayed upon agraphic screen. The present paper attempts to discriminate surface mining features of two such mineral ores, namely Hematite and Uranium. Exploration of geo-statistical tests, i.e. Technique for

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 193–199, 2018.
https://doi.org/10.1007/978-3-319-74690-6_19

Order of Preference by Similarity to Ideal Solution (TOPSIS) and pair-wise Student's t-Test on the pixel values that were derived from the satellite data acquired from three different sensors, viz. Hyperion, ETM+ and LISS III, pertaining to these mineral ores were used as inputs. An Index image was also prepared to separate the signals from mining area and its surroundings.

2 Study Area

Study area comprises of four opencast mining sites, two each pertaining to Hematite, viz. Gua mines situated in the State of Jharkhand, India (at latitude 22.2095°N and longitude 85.3764°E) and Barbil mines situated in the State of Orissa, India (at latitude 22.006°N and longitude 86.45°E) and Uranium, viz. Banduhurang mines situated in the State of Jharkhand, India (at latitude 22.7425°N and longitude 86.1714°E) and Mines d'Arlit, Niger (at latitude 18.45°N and longitude 7.03°E); their geographical locations are depicted here in (see Fig. 1).

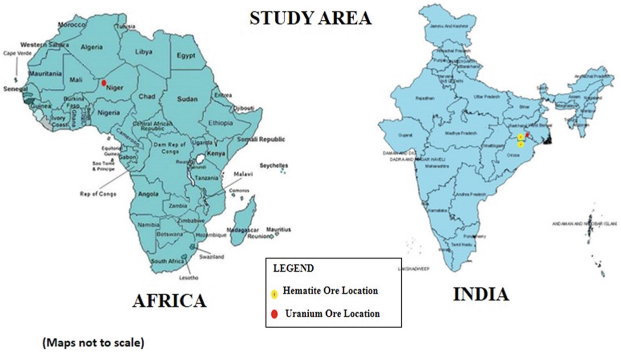


Fig. 1. Locations of study area.

3 Methodology

Satellite data that were procured as freeware had different geographical projection systems; they all were thus transformed to a common World Geodetic System (WGS84). The required pre-processing of hyper-spectral data, obtained from Hyperion, were performed through FLAASH for removing aerosol effects of the atmosphere, MNF for removal of noise from the data, PPI for selecting the pure pixels and SAM in order to match the spectral similarity between an image pixel spectrum and a referenced spectrum. ETM+ data had line-striping; they were thus restored through standard method. Radiometric, geometric and haze corrections were performed over LISSIII data.

A multi criteria decision making screening tool, TOPSIS, was then performed on Hyperion data, pertaining to Hematite, in order to identify and select similar pixels; the same was performed on total 20 pixels, 3 from Gua mines and the remaining 17 from

Barbil mines, the former being duly ground-truthed and hence used as reference for achieving derived result as per the standard procedures [1]. Each pixel location was termed as ‘Profile’; the pixel values are from three identified bands (B#), viz. Band numbers 182, 165 and 164 as detailed out under Tables 1a and 1b.

Table 1a. Pixel values of the referenced Hematite sites from Gua mines.

Pixel location	Lat. (N)	Long. (E)	B#182	B#165	B#164
Profile1	22°4′5.74″	85°26′9.82″	1429	3142	1932
Profile2	22°3′58.93″	85°26′11.98″	1231	1637	1480
Profile3	22°12′43.05″	85°22′27.88″	2347	2069	2330

Table 1b. Pixel values of the Hematite sites from Barbil mines.

Pixel location	Lat. (N)	Long. (E)	B#182	B#165	B#164
Profile1	22°8′23.3918″	85°23′28.8437″	1729	1574	1841
Profile2	21°58′55.378″	85°18′25.7841″	2229	2267	2155
Profile3	21°57′52.8044″	85°19′12.8008″	2623	2108	2268
Profile4	21°55′26.5636″	85°18′53.1899″	3558	3267	3003
Profile5	21°55′11.286″	85°19′53.0464″	1592	1663	1440
Profile6	21°53′20.9039″	85°17′19.7284″	2444	2330	2376
Profile7	21°52′10.2994″	85°18′25.9047″	1848	1631	1893
Profile8	21°50′50.6258″	85°18′21.7066″	556	1198	1090
Profile9	22°14′18.4902″	85°23′0.4970″	2684	2077	2318
Profile10	22°6′37.5901″	85°28′25.4948″	2625	3285	2437
Profile11	22°6′2.0428″	85°27′4.0516″	4094	2950	3399
Profile12	22°9′56.1862″	85°27′57.7158″	2938	3218	2854
Profile13	22°10′5.0000″	85°29′31.0150″	2625	3419	2116
Profile14	22°12′2.7484″	85°31′19.8025″	594	2748	1539
Profile15	21°53′44.3427″	85°25′12.1218″	2157	1341	1507
Profile16	21°51′40.8228″	85°25′12.1218″	3407	4759	3399
Profile17	22°4′0.91″	85°26′15.1″	1406	2671	1662

Virtual removal of vegetation from the LISS III data, pertaining to images of the Indian sites, was done using NDVI in order to equate them with that from the Niger site. Pixel values, pertaining to both Hematite and Uranium, were used to perform pair-wise Student’s t-Test in order to discriminate these two mineral ores from each other [2], for which spectrally comparable data from LISS III and ETM+ sensors were used. In all, 80 pairs of Hematite with Uranium were tested; the pixel values, used for this purpose, are given in Table 2.

Indices or Band Ratio, as they are called, are a part of multiband transformation techniques. Indices are best used for identification of the selected features from a satellite image based on the band statistics [3]. A number of multi-band separability

Table 2. Pixel values used for the pair-wise Student's t-test (B# represents pixel value of the respective bands).

Mining sites	Lat. (N)	Long. (E)	B2	B3	B4	B5
Gua	22°12′44.16″	85°21′51.89″	08	11	18	44
	22°12′36.5″	85°21′20.18″	17	28	32	44
	22°12′17.77″	85°20′58.46″	0	0	34	17
	22°12′43.05″	85°22′27.88″	03	0	48	26
	22°43′51.96″	86°10′43.88″	12	14	29	30
Banduhurang	22°44′2.47″	86°10′16.52″	12	16	29	26
	22°43′48.79″	86°10′6.48″	31	43	52	54
	22°43′33.74″	86°10′30.20″	12	18	35	33
	22°43′53.35″	86°10′30.65″	14	14	24	23
	22°4′5.74″	85°26′9.82″	2	0	52	22
Barbil	23°3′58.93″	85°26′11.98″	3	0	53	25
	22°4′0.91″	85°26′15.1″	7	2	61	31
	22°6′2.04″	85°27′4.05″	4	0	63	26
	7°20′55.9945″	18°46′44.631″	13	99	79	95
Mines d'Arlit	7°21′28.5879″	18°46′9.5675″	63	165	114	136
	7°21′12.1788″	18°46′34.2922″	46	135	95	106
	7°19′45.6379″	18°44′45.5037″	2	71	61	96
	7°18′36.4052″	18°44′49.9991″	4	91	78	120

indices thus may be derived. Each of these indices quantifies, on the basis for the user-defined multi-band statistics, the degree of inter-class separability [4]. Indices are used extensively in mineral exploration and vegetation analyses to bring out small differences between various rock types and vegetation classes [5–7]. A band ratio was generated here to derive indexed images, in order to separate Uranium mining areas from their respective existing surroundings, from the satellite images pertaining to Banduhurang and Mines d'Arlit. Summary of the methodology is depicted in Fig. 2.

4 Results

Results of the present study are summarized as here under:

4.1 The derived result of identification of similar pixel values through TOPSIS, as per the standard procedure, came out to be as high as 82.35% when tested upon Hematite ore.

4.2 The pair-wise Student's t-Test discriminated Hematite and Uranium to the extent of 100% in case of 3 out of 4 sites (i.e., 75%) and 60% in case of the remaining 1 (i.e., 25%), thus averaging to 90% (Table 3).

4.3 Indexed images of Uranium, at both the sites, although separated out the mining areas from their respective surroundings encircled in the figure, (Fig. 3) also extracted many other unidentifiable features.

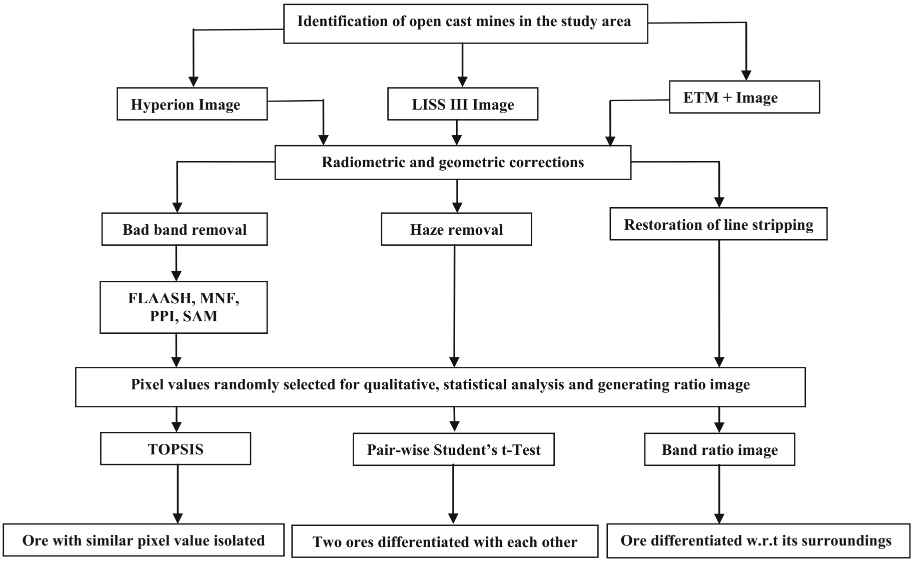

Identification of open cast mines in the study area
Hyperion Image
LISS III Image
ETM + Image
Radiometric and geometric corrections
Bad band removal
Haze removal
Restoration of line stripping
FLAASH, MNF, PPI, SAM
Pixel values randomly selected for qualitative, statistical analysis and generating ratio image
TOPSIS
Pair-wise Student's t-Test
Band ratio image
Ore with similar pixel value isolated
Two ores differentiated with each other
Ore differentiated w.r.t its surroundings

Fig. 2. Methodology.

Table 3. Result of pair-wise Student's t-Test.

Pairs between mineral ores	Pairs tested between locations	No. of pairs tested	Pairs having $t_{cal} > t_{table}$ at 95% confidence level	% of pairs discriminated	Overall discrimination among the mineral ores
Hematite with Uranium	Pixels values from Barbil (Hematite ore) with Banduhurang (Uranium ore)	20	12	60	90%
	Pixels values from Gua (Hematite ore) Banduhurang (Uranium ore)	20	20	100	
	Pixels values from Barbil (Hematite ore) with Mines d'Arlit (Uranium ore)	20	20	100	
	Pixels values from Gua (Hematite ore) with Mines d'Arlit (Uranium ore)	20	20	100	

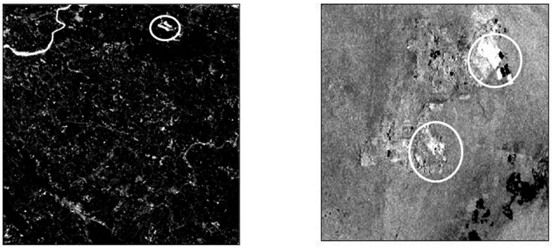


Fig. 3. Indexed images created from the spectral profiles; the mining sites are encircled (Banduhurang at left, Mines d'Arlit at right).

5 Discussion and Future Scope

Correctness in the result of TOPSIS, in identifying similar pixels to the one found from ground-truth in the Hematite mining area, came to be as high as 82.35%, i.e., 14 out of 17 locations. The same is adjudged much better than a similar study conducted by this author earlier [1] when the result achieved remained at the level of 68%; the same could be attributed to higher spectral resolution of the sensor used here (Hyperion in lieu of LISS III in the earlier case) and to the rigorous pre-processing of the satellite data.

Further, the two different mineral ores, Hematite and Uranium, were successfully discriminated from each other using the pair-wise Student's t-Test; average result came to be 90%, wherein in case of 3 out of 4 sites (i.e., 75%) it was as high as cent percent, while in case of the remaining 1 (i.e., 25%) it was merely 60%. The same again is adjudged to have highly been improved upon when compared to author's last efforts to this count when the result was as poor as merely 11% [2].

Spectral Profiles of Uranium at Banduhurang and Mines d'Arlit were found to be similar in satellite images. Hence a single Index could be generated for the regions. However, this index could separate the Uranium mining areas along with various other unidentifiable features that may only be verified through field visits. In the image of Bunduhurang, a river has been also identified on the top left corner. It could be further investigated whether it is a resultant of the radioactive mineral being spilled over to those surrounding areas. The same are left to future investigations.

Hyper-spectral imagery has fine spectral resolution. As such, there is a tremendous scope to explore the possibility of detecting any radioactive phenomenon on earth through satellite data, using hyper-spectral imagery, by developing a universal index for radioactive elements using, geostatistics and advance computing techniques that may include sites of nuclear reactors, radioactive waste dump or and any other source of radioactivity.

Acknowledgement. The authors are thankful to NRSC, India and USGS, USA for providing the satellite data used in the present study.

References

1. Kaur, P., Sharma, R., Mahanti, N.C., Singh, A.K.: Exploration of TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) as an alternative to traditional classification algorithm in small areas of Lohardaga district of Jharkhand, India, using remote sensing image – a case study. Res. J. Earth Sci. **1**(2), 81–85 (2009). ISSN 1995-9044© IDOSI Publications
2. Sharma, R.N.K., Bhatnagar, R., Singh, A.K.: Surface mining signal discrimination using LANDSAT TM sensor: an empirical approach. In: Ell Hassanien, A., et al. (eds.) AMLTA 2012. CCIS, vol. 322, pp. 222–233. Springer, Heidelberg (2012)
3. Wang, J.: Evaluation of lineament detection algorithms using multi-band remote sensing images. In: International Archives of Photogrammetry and Remote Sensing, Vienna, vol. XXXI, part B7 (1996)
4. Thomas, I.L., Ching, N.P., Benning, V.M., D'Aguanno, J.A.: Review article: a review of multi-channel indices of class separability. Int. J. Remote Sens. **8**(3), 331–350 (1987)
5. Weissbeod, T., Karcz, I., Abed, A.: Discussion on the supposed Precambrian palaeosuture along the Dead Sea Rift. J. Geol. Soc. **142**(3), 527 (1988)
6. Cappaccioni, B., Vaselli, O., Moretti, E., Tassi, F., Franchi, R.: The origin of thermal water from the eastern flank of the Dead Rift Valley. Terra Nova **15**(3), 145 (2003)
7. Edgardo, G., James, J.H.: Laser remote sensing of forest and crops in genetic-rich tropical areas. In: International Archives of Photogrammetry and Remote Sensing, ISPRS, vol. XXIX, p. 7 (1992)

Detecting Cross-Site Scripting Attacks Using Machine Learning

Fawaz A. Mereani[1,2](✉) and Jacob M. Howe[1](✉)

[1] Department of Computer Science,
City, University of London, London, UK
{fawaz.mereani,j.m.howe}@city.ac.uk
[2] Umm Al-Qura University, Mecca, Saudi Arabia

Abstract. Cross-site scripting (XSS) is one of the most frequently occurring types of attacks on web applications, hence is of importance in information security. XSS is where the attacker injects malicious code, typically JavaScript, into the web application in order to be executed in the user's browser. Identifying that a script is malicious is an important part of the defence of a web application. This paper investigates using SVM, k-NN and Random Forests to detect and limit these attacks, whether known or unknown, by building classifiers for JavaScript code. It demonstrated that using an interesting feature set combining language syntax and behavioural features results in classifiers that give high accuracy and precision on large real world data sets without restricting attention only to obfuscation.

Keywords: Cross-site scripting · System security
Supervised learning · Classifiers · Features selection

1 Introduction

Web applications are used everywhere and involve sensitive and personal data. This makes them a target for malware exploiting vulnerabilities to obtain unauthorized data stored on the computer. Such attacks include SQL injection, cross-site scripting (XSS) and more. XSS can affect the victim by stealing cookies, modifying a web page, capturing clipboard contents, keylogging, port scanning, dynamic downloads and other attacks [17]. Therefore, the safety of web applications is a very important task for developers. The lack of verification of the client input or the environment is the most common security weakness in web applications [18] and such weaknesses are repeatedly discovered and exploited on both client side and server side. SQL injection and XSS remain in the top ten vulnerabilities listed in the Open Web Application Security Project (OWASP) [14]. This paper investigates the use of machine learning techniques to build classifiers to allow the detection of XSS in JavaScript. The current research focuses on stored or persistent XSS, where a malicious script is injected into a web application and stored in the database. Then on every visit to the page, the script will

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 200–210, 2018.
https://doi.org/10.1007/978-3-319-74690-6_20

be executed on the user's browser. Such an attack might target blogs, forums, comments or profiles [7,11,24].

Work on detecting and protecting against XSS attacks can be broadly categorised into three kinds. Firstly, static analyses which review the source code without execution; whilst such approaches can give formal guarantees that certain vulnerabilities do not occur, they may also be slow or fail to give a result at all. Secondly, dynamic analyses which attempt to determine what the script does at execution time. Modifying the interpreter [16] or checking the syntactic structure [19] are strategies of this analysis. However, it is hard to modify a language's interpreter; vulnerabilities that are caused by the interaction of multiple modules [3] are, therefore, hard to prevent. Thirdly, machine learning can use knowledge of available scripts to build classifiers to predict aspects of the behaviour of new scripts [4]. The advantages of using machine learning are: first, once a classifier has been built it can quickly predict whether or not a script is malicious; second, it does not need a sandbox to analyse the script; third, the classifier has predictive capabilities to detect new malicious JavaScript. In this work, machine learning is used to detect stored XSS with high accuracy and precision. Scripts may well be obfuscated; importantly, the aim is to classify all scripts, whether obfuscated or not. The design space for such an approach is large, with choices of how to build classifiers, and an even larger choice of how to abstract concrete code into a collection of features that the machine learning algorithms will work on. The contributions of this work are as follows:

- a new selection of program features, drawn from program syntax and program behaviours, is given for the learning algorithms to work on
- the collection of a balanced dataset of scripts from multiple sources giving good coverage of both malicious and benign scripts
- the use of support vector machines (SVM), k-nearest neighbour (k-NN), and Random Forests as learning algorithms to give classifiers; this is the first evaluation of Random Forests on XSS problems
- the evaluation of the resulting classifiers on training and real world data.

The rest of this paper is organised as follows: Sect. 2 gives an overview of relevant aspects of JavaScript and JavaScript obfuscation. Section 3 discusses related work on machine learning and XSS. Section 4 details the dataset collection and features selection. Section 5 gives the experimental data on the performance of the classifiers, and includes discussion of the results. Further discussion, direction of future work and conclusions are given in Sect. 6.

2 Background

2.1 JavaScript

JavaScript is a language commonly used in the development of web pages to make them more dynamic and interactive. It is client-side which allows the source code to be executed in the web browser rather than on the server. This allows functions to run after loading the web page without the need to communicate with the server, for example, producing an error alert before sending information to the server. Scripts can be inserted within the HTML or can be referenced in a separate .js file. JavaScript is a good choice for attackers to carry out their attacks and to spread them over the Internet, because the majority of websites use JavaScript and it is supported by all web browsers. Hence, it is the target of many XSS, SQL injection and passive download attacks [22].

2.2 Obfuscation

The goal of obfuscation is to modify the code to make it hard to read or understand. For example, by changing the names of variables or functions, or by using operators to compound terms to give program constructs. Both benign and malicious scripts can use obfuscation techniques with different purposes for each one. Benign obfuscation aims to protect privacy or intellectual rights, while malicious obfuscation works on disguising malicious intentions and evading the static inspection checks. Multiple obfuscation methods can be applied by attackers to best hide malicious scripts [25]. A simple example of malicious JavaScript obfuscation by using URL Encoded is:

%3Cscript%3E%0D%0Aalert%28document.cookie%29%3B%0D%0A
%3C%2Fscript%3E

The original script after deobfuscated is as follows:

$$< script > alert(document.cookie); < /script >$$

This paper considers JavaScript that may or may not be obfuscated and aims to classify scripts as either malicious or benign in either case.

3 Related Work

A number of approaches have been taken to dealing with XSS. The standard approach for the web application developer is to use sanitization and escaping to prevent untrusted content being interpreted as code [23,24]. Alternatively parser-level isolation can confine user input data during the lifetime of the web application [12]. Note that this isn't detection of XSS, rather prevention of its execution through good coding practice. This is preferred to blacklists which are viewed as easy to circumvent [24]. Another technique to defend against XSS vulnerability is to use randomized namespace prefixes with primitive markup language elements to make it hard for the attacker to use these elements [20]. Previous methods aim to remove malicious elements from untrusted data, however, as with blacklists some XSS vectors can easily bypass many powerful filters. In [8] rules are generated to allow control of communications, with a web proxy blocking communication with untrusted sites. Combinations of static and dynamic techniques use taint analysis to prevent sensitive data being sent to a third party by monitoring the flow of data in the browser [21].

Machine learning techniques have been applied to detecting XSS attacks [10] and are attractive because they can adapt to changes and variations in malicious scripts [9]. Likarish et al. [10] evaluated Naive Bayes, ADTree, SVM, and RIPPER classifiers in detecting obfuscation of scripts (as a proxy for malicious), using features that track the number of times symbols appears in benign and malicious scripts. The classifiers were evaluated using 10-fold cross validation giving precision of 0.92. It should be noted that the test set of obfuscated scripts is small. The approach of Likarish et al. was expanded by Nunan et al. [13], where features were categorized into three groups: (1) obfuscation based, (2) suspicious patterns and (3) HTML/JavaScript schemes. Naive Bayes and Support Vector Machine classifiers were used to classify scripts as XSS or non-XSS. Three datasets were used, malicious (obfuscated) scripts from XSSed.com and benign scripts from Dmoz and ClueWeb09. The classifiers were evaluated using accuracy to give 98.58% with Dmoz dataset, and 99.89% with the ClueWeb09 dataset. This approach has high accuracy, but depends on a single source for malicious scripts and again focuses on obfuscated scripts. Another study analyzing malicious scripts and feature extraction was conducted by Wang et al. [22] where the main idea of feature extraction is that some functions are of limited use in the benign scripts, but are used much more in malicious scripts, such as the DOM-modifying functions, the eval function, the escape function. This technique gives accuracy of up to 94.38%. However, again the technique concentrates only on obfuscated scripts and on DOM-modifying functions. The work of [2] also aims to distinguish between obfuscated and non-obfuscated scripts. Their method gives high precision results up to 100% though again the number of malicious scripts used was small. Komiya et al. [9] used machine learning techniques to classify user input to detect malicious web code. Feature extraction depended on two methods, blank separation, and tokenizing. The idea of the first method is that input contains many terms separated by spaces, a count of each term is used for calculation of feature weight. It should be noted that in a malicious script terms might be separated by characters other than spaces, which would lead to an incorrect feature weight. The second method is based on the idea that malicious code contains tokens that describe the features of malicious web code, with a count of each term used to calculate feature weights. Using this feature extraction technique with SVM gave accuracy of up to 98.95%.

4 Methodology

4.1 Datasets

This paper concentrates on malicious and benign scripts that can be sent to Web applications via HTTP requests. The attacker can use obfuscated scripts, as well as scripts written in the normal manner. To create balanced datasets JavaScript was collected from a number of trusted sources including both obfuscated and non-obfuscated scripts and scripts of with a variety of lengths. Two datasets were gathered. The first data set was collected for training and the second for testing. There is some overlap in the sources of the scripts, but not

Table 1. Structural features

Features Group	**Terms**
Punctuation	&,%, /, \, +, ’, ?, !, ;, #, =, [,], $, (,), ∧ *, , , , -, <, >, @, _ , :, {, }, ˜, ., space, |, ¦, ”
Punctuation Combinations	><, ′ ” ><, [], ==, &#

in the scripts themselves. The benign scripts were obtained from a number of developer and university sites.

For the training set, malicious scripts were obtained from developer sites [1,6,15], a selection from XSSed, the largest online archive of XSS vulnerable websites [5] and additional scripts were collected by crawling sites known to be untrustworthy. The test set was drawn entirely from the XSSed archive. Again there is no overlap between sets. The first (training) dataset contains 2000 of each of malicious and benign scripts. The second (test) dataset contains 13,000 each of malicious and benign scripts. Data was prepared for the classification experiments by removing duplicates to get unique scripts, removing extra blank spaces and unnecessary new lines, and lowercasing all letters.

4.2 Selecting Features

There is a large design space for selecting suitable features of JavaScript in order to start classifying scripts. Features in this work are categorized into two groups, (1) structural, and (2) behavioural. In total, 59 features are considered.

Structural Features. The structural features are the complete set of non-alphanumeric characters that can occur in JavaScript. These may occur in any script, but if the attacker is using techniques to trick the protection on Web applications this can change the range of characters used in a script. This applies whether or not the script is obfuscated. To give a simple example, a malicious script might add spaces or unnecessary symbols between commands or tags, such as $< \backslash\ sc\ ri\ pt >$. A benign script would not do this. As another example consider a cookie access separated into two parts and the use of the + sign to recombine the entire command again, $document +' .' + cookie$. Also included in the structural features are combinations of characters that might be used in constructing malicious scripts. There are 33 non-alphanumeric characters, and 5 further combinations of these are considered. The features might be measured in a variety of ways. In the current work the measure is a 0/1 value indicating that the feature does not or does occur in the script. As will be demonstrated later this surprisingly simple measure works very well. Table 1 gives the structural features (where space indicates the blank space character).

Behavioural Features. These are a selection of the commands and functions that can be used in JavaScript. The attacker may use them suspiciously and differently from the benign developer. That is, the benign developer does not

Table 2. Behavioural features

Features	Description
Readability	Is the script readable - the number of alphabetical characters
Objects	Document, window, iframe, location, This
Events	Onload, Onerror
Methods	createelement, String.fromCharCode, Search
Tags	DIV, IMG, <script
Attributes	SRC, Href, Cookie
Reserve	Var
Functions	eval()
Protocol	HTTP
External File	.js file

need to hide the intent of their code, whilst on the contrary, the attacker will use a range of commands to create the malicious script. For example, using the eval function frequently, using de-obfuscated functions in the script, or including a malicious script within an image tag. The insight is that combinations of occurrences of commands indicate suspicious activity. There are 21 of these consider in this work. As for the structural features, behavioural features might be measured in many ways. The current work again uses a 0/1 value indicating that the feature does not or does occur in the script. Table 2 gives the behavioural features selected for their potential use in malicious scripts.

4.3 Classifiers

The feature data is used as input for supervised learning algorithms. In this work, support vector machines (SVM), k-nearest neighbour (k-NN), and Random Forests are used, although other classifiers might also be used. Two variations on SVMs are used, with a linear kernal and with a polynomial kernal. A number of parameters used with SVMs were tuned during the training phase: BoxConstraint to control the maximum penalty of misclassification, and OutlierFraction to determine the expected proportion of outliers in the training data. For the k-NN classifier parameter k, the number of neighbours was tuned. For the Random Forest classifier the number of trees in the forest was tuned.

5 Results

5.1 Experiments

MatLab R2016b was used for the experimentation. The experiments focused on the performance of SVM, k-NN, and Random Forest classifiers using the datasets and features described in Sect. 4. For the first set of results the training dataset was divided at random into five folds, with training on four of the five folds,

Table 3. SVM (Linear Kernel) evaluation

Folds	Accuracy	Precision	Sensitivity	Specificity
1st	94.93%	93.98%	96.37%	93.40%
2nd	94.75%	93.19%	95.69%	93.92%
3rd	95.06%	94.66%	95.03%	95.08%
4th	94.14%	93.14%	96.63%	93.34%
5th	94.81%	93.25%	96.25%	93.45%
Average	94.74%	93.64%	95.99%	93.84%

Table 4. SVM (Polynomial Kernel) evaluation

Folds	Accuracy	Precision	Sensitivity	Specificity
1st	96.87%	96.10%	97.95%	95.70%
2nd	96.81%	96.20%	97.09%	96.55%
3rd	97.43%	98.04%	96.66%	98.14%
4th	96.87%	96.25%	97.47%	63.25%
5th	97.31%	96.75%	97.85%	96.78%
Average	97.06%	96.67%	97.40%	96.68%

Table 5. k-NN classifier evaluation

Folds	Accuracy	Precision	Sensitivity	Specificity
1st	97.00%	96.34%	97.96%	95.95%
2nd	96.50%	96.20%	96.45%	96.53%
3rd	97.43%	97.65%	97.02%	97.82%
4th	97.75%	97.38%	98.11%	97.38%
5th	96.93%	96.37%	97.47%	96.41%
Average	97.12%	96.79%	97.40%	96.82%

and testing on the remaining fold. This five fold testing then gives five training experiments. The SVM with linear kernel was tuned to set the BoxConstraint parameter to 7. The polynomial kernel was tuned by setting the OutlierFraction parameter to 0.10. k-NN was tuned by setting NumNeighbors parameter to 1 (since some malicious scripts might be singletons). Random Forest was tuned by setting the number of tree to 40. The results are described with Precision (often called Detection Rate in a security context), Accuracy, Sensitivity and Specificity. Table 3 shows results with test data for SVM with linear kernel, Table 4 shows the results for SVM with polynomial kernel, Table 5 shows the results for k-NN, and Table 6 shows the results for Random Forest.

To test real world attacks, models for SVM with both linear and polynomial kernel, k-NN, and Random Forest were built by training classifiers using the whole training dataset. Then the testing dataset that contains new malicious and benign scripts was used for testing the classifiers' performance. In Table 8 the results of this test are given and in Table 7 the confusion matrices giving the raw data are presented.

Table 6. Random Forest classifier evaluation

Folds	Accuracy	Precision	Sensitivity	Specificity
1st	96.43%	95.28%	97.93%	94.83%
2nd	96.75%	95.54%	97.59%	96.00%
3rd	97.81%	97.65%	97.78%	97.83%
4th	97.68%	97.00%	98.35%	97.03%
5th	97.43%	97.00%	97.85%	97.02%
Average	97.22%	96.47%	97.90%	96.54%

Table 7. Confusion matrix with testing data

	Linear		Polynomial		k-NN		Random Forest	
	Malicious	Benign	Malicious	Benign	Malicious	Benign	Malicious	Benign
Malicious	12783	217	12960	40	12985	15	12980	20
Benign	739	12261	62	12938	50	12950	110	12890

5.2 Discussion

The experiments give two sets of data. The first uses a training set of scripts chosen to give coverage of a variety of styles of scripts – obfuscated or not, varying length. The five fold evaluation shows good performance for the classifiers, with (as expected) SVM with a polynomial kernal giving stronger results than for SVM with linear kernal. The second set of data is designed to give a real-world evaluation of the classifiers learnt from the entire training set. As can be

Table 8. Evaluation with testing data

	Linear	Polynomial	k-NN	Random Forest
Accuracy rate	96.32%	99.60%	99.75%	99.50%
Precision rate	98.33%	99.69%	99.88%	99.84%
Sensitivity (TPR)	94.53%	99.22%	99.61%	99.15%
Specificity (TNR)	98.26%	99.69%	99.88%	99.84%

observed in Table 8 the SVM, k-NN, and Random Forest classifiers can distinguish between malicious and benign scripts with high accuracy and detection rate. k-NN performs marginally better than SVM and Random Forest, with accuracy of 99.75% and precision 99.88%. The confusion matrices of Table 7 show the small numbers of false positives and failed detections, 15 of the former and 50 of the latter for the k-NN classifier. These results suggest that classifier based techniques can be a powerful tool for detecting XSS attacks.

6 Conclusion

This paper has demonstrated that SVM, k-NN, and Random Forest can be used to build classifiers for XSS coded in JavaScript giving high accuracy (up to 99.75%) and precision (up to 99.88%) when applied to a large real world data set. This shows that these classifiers can be added as a security layer either in a browser or (as intended) on a server. The training data was designed to give fair coverage of scripts, including scripts of a variety of lengths and both obfuscated and non-obfuscated scripts. The data is labeled as malicious or benign, rather than using obfuscation as a proxy for maliciousness. Whilst SVM, k-NN, and Random Forest have been used in the experiments, it is expected that other classification methods would also work well.

A systematic direct comparison with previous studies is not possible, however, the new classifiers give performance statistics that stand up well. The current study works with a larger and more diverse suite of scripts than many of these previous studies, and is the first study to use Random Forests as a classifier for XSS. The key to building successful classifiers is the choice of feature set and how the features are measured. With a large design space there is motivation to investigate a wide range of approaches to feature selection. The features chosen in this paper fall into two categories: firstly the complete set of symbols used in the JavaScript language, and secondly aspects of the scripts that are associated with malicious code. This allows the classifiers to find patterns based on the shape of the program (symbols) and the constructs used (behavioural features). One particularly interesting aspect of this work is that, in contrast to other studies, a binary measure has been used for all features. This has given higher accuracy and precision than earlier experiments using weighted measures. This hints that it may be possible to extract rules from the classifiers that describe malicious scripts. Another interesting aspect is the value of k used in the final experiments is 1. This suggests that malicious scripts might well be singletons that stand apart from clusters of benign scripts. Future work is to investigate these aspects, as well as to use the same features with a Neural Network classifier.

References

1. Examples of malicious javascript (2014). https://aw-snap.info/articles/js-examples.php. Accessed 16 Dec 2016
2. Aebersold, S., Kryszczuk, K., Paganoni, S., Tellenbach, B., Trowbridge, T.: Detecting obfuscated JavaScripts using machine learning. In: International Conference on Internet Monitoring and Protection. IARIA Press (2016)
3. Balzarotti, D., Cova, M., Felmetsger, V., Vigna, G.: Multi-module vulnerability analysis of web-based applications. In: Computer and Communications Security, pp. 25–35. ACM Press (2007)
4. Domingos, P.: A few useful things to know about machine learning. Commun. ACM **55**(10), 78–87 (2012)
5. Fernandez, K., Pagkalos, D.: XSS (Cross-Site Scripting) information and vulnerable websites archive. XSSed.com. Accessed 14 June 2017
6. Karnad, K.: XSS payloads you may need as a pen-tester (2014). https://www.linkedin.com/pulse/20140812222156-79939846-xss-vectors-you-may-need-as-a-pen-tester. Accessed 25 Dec 2016
7. Kirda, E., Jovanovic, N., Kruegel, C., Vigna, G.: Client-side cross-site scripting protection. Comput. Secur. **28**(7), 592–604 (2009)
8. Kirda, E., Kruegel, C., Vigna, G., Jovanovic, N.: Noxes: a client-side solution for mitigating cross-site scripting attacks. In: Symposium on Applied Computing, pp. 330–337. ACM Press (2006)
9. Komiya, R., Paik, I., Hisada, M.: Classification of malicious web code by machine learning. In: Awareness Science & Technology (iCAST), pp. 406–411. IEEE (2011)
10. Likarish, P., Jung, E., Jo, I.: Obfuscated malicious JavaScript detection using classification techniques. In: Malicious and Unwanted Software (MALWARE), pp. 47–54. IEEE (2009)
11. Malviya, V.K., Saurav, S., Gupta, A.: On security issues in web applications through cross site scripting (XSS). In: Asia-Pacific Software Engineering Conference, vol. 1, pp. 583–588. IEEE (2013)
12. Nadji, Y., Saxena, P., Song, D.: Document structure integrity: a robust basis for cross-site scripting defense. In: Network and Distributed System Security Symposium. Internet Society (2009)
13. Nunan, A.E., Souto, E., dos Santos, E.M., Feitosa, E.: Automatic classification of cross-site scripting in web pages using document-based and url-based features. In: Computers and Communications, pp. 702–707. IEEE (2012)
14. OWASP Top 10 - 2017 rc1 (2017). https://www.owasp.org. Accessed 7 June 2017
15. XSS Payloads: XSS payloads you may need as a pen-tester. http://www.xss-payloads.com/payloads.html. Accessed 14 Oct 2016
16. Pietraszek, T., Berghe, C.V.: Defending against injection attacks through context-sensitive string evaluation. In: Recent Advances in Intrusion Detection, Lecture Notes in Computer Science, vol. 3858, pp. 124–145. Springer (2005)
17. Raman, P.: JaSPIn: JavaScript based anomaly detection of cross-site scripting attacks. Ph.D. thesis, Carleton University, Ottawa (2008)
18. Rocha, T.S., Souto, E.: ETSSDetector: a tool to automatically detect cross-site scripting vulnerabilities. In: Network Computing and Applications, pp. 306–309. IEEE (2014)
19. Su, Z., Wassermann, G.: The essence of command injection attacks in web applications. ACM SIGPLAN Not. **41**(1), 372–382 (2006)

20. Van Gundy, M., Chen, H.: Noncespaces: using randomization to defeat cross-site scripting attacks. Comput. Secur. **31**(4), 612–628 (2012)
21. Vogt, P., Nentwich, F., Jovanovic, N., Kirda, E., Kruegel, C., Vigna, G.: Cross site scripting prevention with dynamic data tainting and static analysis. In: Network and Distributed System Security Symposium, p. 12. Internet Society (2007)
22. Wang, W.H., Yin-Jun, L.V., Chen, H.B., Fang, Z.L.: A static malicious javascript detection using SVM. In: International Conference on Computer Science and Electronics Engineering, vol. 40, pp. 21–30. Atlantis Press (2013)
23. Weinberger, J., Saxena, P., Akhawe, D., Finifter, M., Shin, R., Song, D.: A systematic analysis of XSS sanitization in web application frameworks. In: European Symposium on Research in Computer Security. Lecture Notes in Computer Science, vol. 6879, pp. 150–171. Springer (2011)
24. Williams, J., Manico, J., Mattatall, N.: Cross-site Scripting (XSS). https://www.owasp.org/index.php/Cross-site_Scripting_(XSS). Accessed 22 July 2016
25. Xu, W., Zhang, F., Zhu, S.: JStill: mostly static detection of obfuscated malicious JavaScript code. In: Data and Application Security and Privacy, pp. 117–128. ACM Press (2013)

Text Mining Approach to Extract Associations Between Obesity and Arabic Herbal Plants

Samar Anbarkhan(✉), Clare Stanier(✉), and Bernadette Sharp(✉)

Staffordshire University, College Road, Stoke-on-Trent, ST4 2DE, UK
samar.anbarkhan@research.staffs.ac.uk,
{c.stanier,b.sharp}@staffs.ac.uk

Abstract. Historical information on herbal medicines is underexploited and this is particularly true of the important resources of Arabic herbal medicines. Current research into Arabic medicinal plants as alternative medicine is limited and there is a lack of accurate translations and interpretations of herbal medicine texts. This research focuses on an investigation of Arabic herbal medicinal plants in relation to the problem of obesity. This paper demonstrates how text mining can help extract relevant concepts associated with Arabic herbal plants and obesity in order to discover associations between the herbal medicinal ingredients and obesity symptoms.

Keywords: Text mining · Association rules · Apriori algorithm · Obesity Arabic herbal medicine

1 Introduction

Global estimates suggest that about 500 million adults are obese worldwide, with prevalence rates rising among children and adolescents [1]. Obesity is a serious problem globally, and in Saudi Arabia in particular. Overweight and obesity rates for adults in this region are estimated at 30.4% and 12% respectively; and in the Gulf countries the rates are reaching as high as 66% and 31.5% respectively [2]. Different drugs are available for weight reduction; however, they sometimes generate adverse toxicities, which may lead to serious side effects. Hamid et al. [3] explains that complementary medicines are now often used when clinical medicine proves unsuccessful and/or pernicious. In many developing countries medical herbal complements are lauded as effective, and are relied on by traditional practitioners, partly for historical and cultural reasons [4]. Despite, herbal medicines were discarded from traditional medical use in the mid-20th century on economic grounds [5].

Historical and current studies indicate that the Mediterranean region has been known for its a rich inventory of complementary alternative medicine, in particular herbal medicine [6]. For Arabic herbal medicine however, no serious efforts have been made to collect and integrate these scattered and thinly-spread resources. In recent years, the aspiration to capture the wisdom of traditional healing systems has revived the interest in herbal plants, in Europe and North America [4] and in other areas such as China [6]. Although traditional Arabic medicine has contributed to modern western medicines, it

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 211–220, 2018.
https://doi.org/10.1007/978-3-319-74690-6_21

is still an underexploited resource and has not emerged as a comprehensive alternative treatment. In spite of its rich inventory, research into the usage of Arabic medicinal plants is limited. Discussions around herbal medicines are found principally in informal literature, often with imprecise and non-technical language [7]. As a result, there are a number of barriers related to the use of herbal treatment in mainstream medicine, partly due to the lack of accurate translations and interpretations of herbal medicine texts, and partly because of lack of training in both modern medicine and herbalist practitioners [8]. According to [9] Arabic medicinal plants are becoming rare because of climatic and environmental changes of their natural habitat. The global increase interest in herbal medicinal plant and the ongoing environmental risk of destruction of the natural habitat of herbal medicinal plants indicate an urgent need to preserve the existing knowledge of these medicinal plants. More research is therefore needed to understand Arabic herbal medicinal plants. Our current efforts are dedicated to the understanding and discovery of Arabic herbal medicinal plants in relation to the epidemic of obesity and linked diseases.

This paper attempts to demonstrate how a text mining approach can help elicit potentially relevant associations between Arabic herbal medicine and the treatment of medical conditions, in this instance, the treatment of obesity. In text mining research the contribution to knowledge is typically the discovery of new knowledge extracted from concepts and relationships identified through the natural language processing stages. The outcomes of the text mining research presented here contribute to the preservation of the body of knowledge about Arabic medicinal plants, and their medicinal usage with respect to obesity and associated diseases.

The paper is structured as follows. Section 1 has discussed the context and motivation of our research. Section 2 provides an introduction to text mining and its applications in the area of health care. Section 3 describes our text mining process which supports this research. Section 4 summarises the main findings and suggests future directions for research.

2 Related Work

Text mining is defined as the process of analysing unstructured text in order to extract useful information using computational linguistics approaches and software tools [10]. It is a sub-field of knowledge discovery which includes data mining, link mining and web mining [11]. It is a multidisciplinary field involving pattern matching, information retrieval and information extraction. Pattern matching is used to analyse text, detect language patterns such as punctuation marks, characters, syllables, words, and phrases [12] whereas information retrieval selects the most relevant documents for further analysis. Information extraction is the task of extracting specific types of information from documents, such as extracting specific entities/concepts such as people, organisations, locations, etc. [13]. Recognition and extraction of these entities is a primary task involving natural language processing algorithms.

Text mining has become one of the core topics in healthcare and medical applications [14]. It is applied to discover new knowledge from medical documents and databases, which may contain implicit information crucial to advancing treatments, medical technology and other healthcare activities. The literature review has revealed a large number of medical applications focusing mostly on cancer and diseases related to genes and proteins [15], but others on testing unknown drug-disease relations [16] and discovering associations between chemical entities [17]. Text mining algorithms include the application of maximum entropy-based named entity to extract relationships between prostate cancer and relevant genes [18]. Other studies focused on identifying strong correlations between Chinese herbal medicine and diseases such as obesity [19], rheumatoid arthritis [20], articular chondrocytes [21], urinary tract infections [22] and age-related dementia [23]. Association rules are applied by [24] to reveal the relationships between diseases and related herbal materials in Chinese medicine while [25] develop a Chinese herbal medicine (CHM) network and core CHM treatments for acne. Other approaches include visualisation techniques applied by [26] and natural language processing and rule base system adopted by Fang et al. [27] to extract traditional Chinese medicine effectors and effects.

3 Research Approach

Our text mining approach involves six main tasks which are described in Fig. 1.

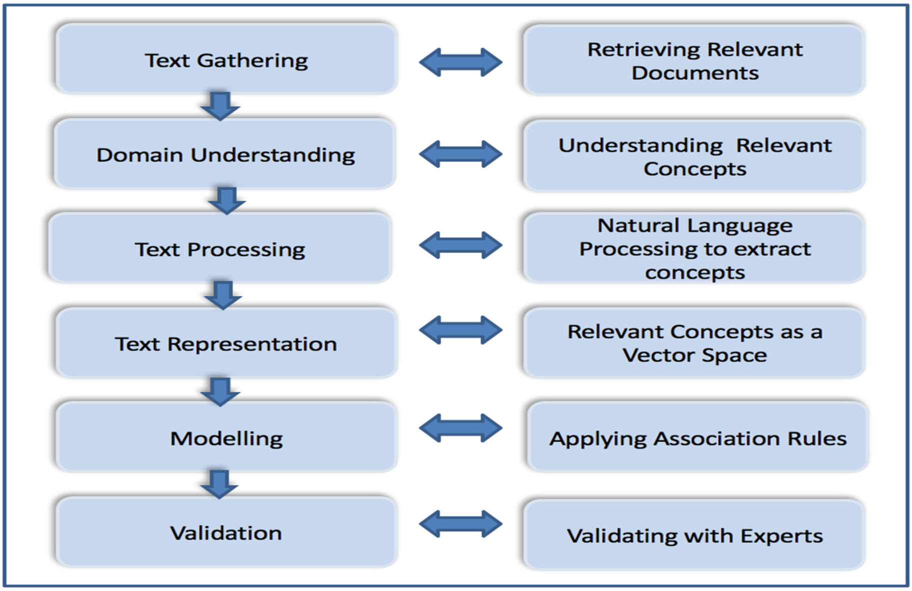

Text Gathering
Retrieving Relevant Documents
Domain Understanding
Understanding Relevant Concepts
Text Processing
Natural Language Processing to extract concepts
Text Representation
Relevant Concepts as a Vector Space
Modelling
Applying Association Rules
Validation
Validating with Experts

Fig. 1. Text mining tasks

Our approach is based on CRISP methodology which is well known data mining methodology. As our dataset is textual and unstructured, we have combined CRISP methodology with natural language processing analysis to extract the important concepts which can be then used as the basis for the modelling and discovery of associations between obesity and Arabic herbal plants concepts.

3.1 Text Gathering

The text gathering stage involved extracting scientific articles relevant to obesity and herbal medicine from medical databases and academic research. For this purpose, bibliographic databases such as PubMed, ScopeMed, Elsevier, ClinMed, PMC, CDC, Hindawi ResearchGate, Scientific Research and Science Publishing Group, PubMed Central were used. The research retrieved only those herbal medicinal plants known in countries which use Arabic as the official language. We have excluded articles discussing imported herbal medicine. Most of the articles reviewed related to our Saudi Arabia, our major focus, and where herbal medicines are commonly used alongside conventional drug therapy.

3.2 Domain Understanding

To test the feasibility of our approach, the initial implementation was based on 30 abstracts related to obesity as retrieved from medical journals namely Preventing Chronic Disease, International Journal of Medical Science and Public, Health and Journal of Obesity & Weight Loss Therapy and 30 journal articles discussing Arabic herbal medicine namely Journal of Medicinal Plants Research, Austin Journal of Nutrition and Food sciences, and Journal of Community Medicine & Health Education. This initial set of 30 herbal medicine articles is considered a large enough corpus to test the feasibility of our approach; this will also provide textual data to investigate and extract the important concepts/features contained in both the herbal medicine and obesity literature.

In this proof of concept stage, the focus is on extracting nouns and noun phrases considered as relevant concepts describing obesity and herbal medicinal plants. The selection of relevant concepts is assisted by medical glossary of obesity terms published in medical websites (e.g. emedicine health, NHS UK) as well as medical dictionaries, whereas the extraction of herbal plants concepts is based on handbooks such as Handbook of Arabian Medicinal Herbs and Prophetic Medicine & Herbalism.

3.3 Text Analysis

This task applies natural language processing steps to extract relevant features from the retrieved medical textual data. The lexical analysis step extracts relevant concepts associated with obesity and Arabic herbal medicinal plants. It involves removing stop-words and terms deemed unhelpful in document classification (e.g. verbs), grouping lexical items in a meaningful way, for example remedy, treatment and medication are grouped into one concept namely medicament, hypertension and high blood pressure are equivalent and thus merged.

The extraction of concepts is carried out in two steps: (i) speech tagging focuses on extracting nouns and nouns phrases related to obesity, and (ii) sematic representation of these concepts. Speech tagging is achieved using the Stanford CoreNLP software, which provides a grammatical analysis toolkit and includes modules such as PoS-taggers, tokenisers, named entity recognisers, syntactic parsers and coreference resolution systems.

The semantic and pragmatic analysis steps provide the context for the building of the semantic representation of the extracted concepts for both topics: obesity and herbal medicine. These extracted concepts are represented using semantic networks which can capture the relationships of these concepts as described in Fig. 2. This process extracted a total of 19 concepts associated with obesity.

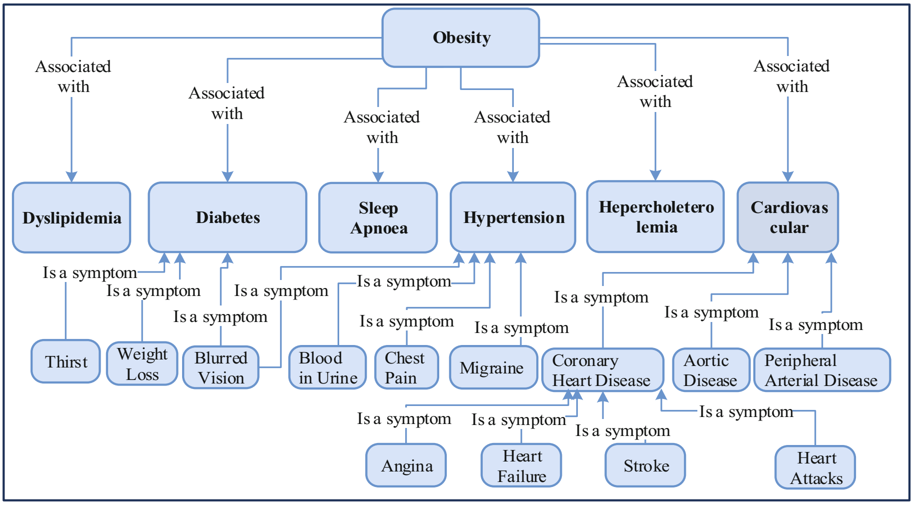


Fig. 2. A semantic network of obesity concepts

Many challenges had to be met in building the semantic representation of these concepts.

- The first challenge is to analyse the terms and descriptions of the unstructured textual dataset to unlock the information. The content is written for a diverse audience; for example, some papers use scientific and medical terms while others use common, non-scientific, terms, for example, hypertension vs. high blood pressure, onion vs. allium cepa. This created a requirement for mapping the symptoms of obesity and associated diseases to higher-level categories and providing a way to standardise the herbal plant concepts
- The second challenge relates to the fact that some herbs have multiple names; for this reason, the Latin or scientific name was used, as shown in Table 1.
- The third challenge is the process of extracting interesting but nontrivial patterns or associations from our textual dataset which is unstructured text.

Table 1. Herb names

Latin name	Common name
Nigella sativa L.	Blackseed, black-caraway, black-cumin, fennel-flower, nigella, nutmeg-flower
Allium sativum L.	Garlic

It was decided to focus exclusively on physical symptoms and conditions associated with obesity and not to consider psychological symptoms and conditions such as depression. This decision was made because the link between physical symptoms and obesity is clearer in the literature and provided a more specific focus on obesity. Mapping to higher-level entities was used to standardise informal descriptions of symptoms and conditions, and to support analysis. For instance, symptom names were identified by medical terms for clarity, and to provide consistency for expert evaluation. Figure 3 shows an example of higher-level entity mapping for the one of the associated obesity concepts, namely cholesterol.

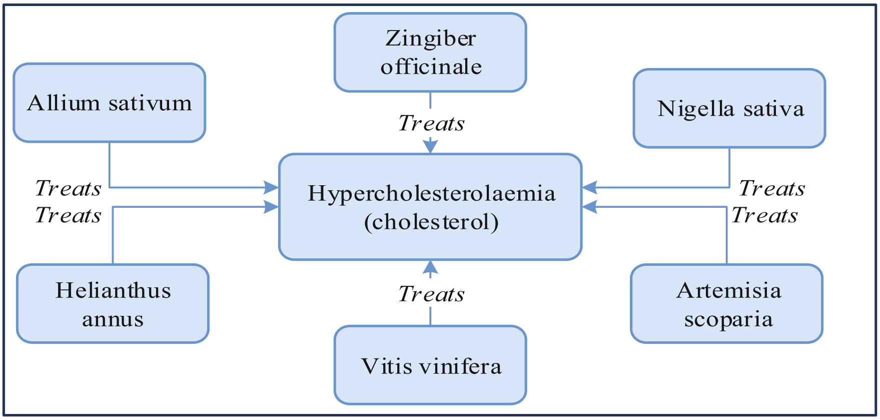


Fig. 3. A higher-level entity of cholesterol

3.4 Text Representation

The concepts extracted from the previous steps are represented in terms of a vector space, to be used for the next modelling step. An example of this vector space is given in Table 2.

Table 2. An example of data representation (F = False, T = true)

Scientific name	Hypercholesterolaemia	Obesity	Diabetes	Hypertension	Cardiovascular
Avena sterilis L.	F	F	T	F	F
Olea europaea L.	F	F	T	F	F

3.5 Modelling

This task focuses on selecting and applying text mining modelling techniques to discover new knowledge. As we are concerned with discovering relationships between herbal medicine and obesity and related diseases, this study selected association rules as the primary modelling algorithm. Three sets of experiments, using WEKA software, were carried out, aimed at extracting any associations between obesity and herbal plants concepts by applying Apriori and predictive Apriori algorithms. Apriori algorithm uses priori knowledge of frequent itemset property for association rule mining satisfying a given minimum support threshold whereas the predictive Apriori is concerned with the predictive accuracy of an association rule. No meaningful associations were produced by the Apriori algorithm. The predictive apriori algorithm has performed better than the Apriori algorithm as it tries to maximise predictive accuracy of an association rule rather than confidence in Apriori. The first experiment was based on 112 herb instances and six diseases producing 20 associations. The second experiment was extended to include 298 herb instances and seven diseases; this has discovered 20 associations. The third experiment, which grouped the herbs by family name generated 81 associations. Best rules are selected based on their confidence. Table 3 lists the association with confidence values above 0.96. Grouping herbs by their family name has produced the best results but the risk is that we may lose potential valuable information. Therefore it may be necessary to drill down to individual herbs to investigate any outliers in our textual data.

Table 3. Association with the highest confidence values using predictive apriori

Associations	Confidence
Trigonella foenum-graecum L. ==> Diabetes	0.98434
Citrullus colocynthis (L.) Schard. ==> Diabetes	0.98121
Allium cepa L. ==> Diabetes	0.97644
Juglans regia L. ==> Diabetes	0.97644

A clustering approach using k-means and expectation maximisation was used to discover the grouping structures of our dataset. However, it failed to reveal any meaningful relationship between clusters and their elements.

3.6 Validation and Discussion

In addition to support and confidence measures, the extracted rules were discussed with a focus group which included 20 domain experts based in Saudi Arabia. This group consisted of one general practitioner, one laboratory specialist, one therapist, one community physician, one fitness team leader, three pharmacists, and 12 clinical nutritionists and dietitians. The aim of the validation was to determine whether the extracted association rules were regarded as credible by these experts, and to investigate their views on herbal medicine. The semantic net representation was also discussed.

The outcomes of the validation were as follows. The associations were found to be correct by 10 experts and partially correct by 2 experts. Six participants did not comment, two respondents thought the associations were not precise. This suggested that useful associations can be derived from literature but that further work is required to refine and clarify the associations. With regards to the semantic net, a few modifications were recommended, for example, the addition of bile stones and additional symptoms for some conditions such as diabetes. Concerns were expressed about the use of herbal medicines by people with little understanding of their side effects and drug interaction. It was established that most of the domain experts interviewed do not use herbs, highlighting the fact that Arabic herbal medicine is still an underexplored resource. Herbal medicine is not offered unless requested, suggesting that use of herbal medicine in a clinical context is declining.

4 Conclusions and Future Work

The biggest challenge with text mining is the challenge implicit in analysing large amounts of unstructured information to extract new patterns. This is compounded by the limited set of tools available to analyse unstructured text directly for mining. Furthermore, there is the challenge of identifying relevant materials due to the volume of data, and the fact that data may not be classified. The findings of this research suggest areas for further work, particularly in relation to the generation and validity of association rules in this context. The text mining approach described in this paper enabled the extraction of important concepts pertinent to herbal medicine and obesity and related diseases. The semantic net representation of the domains allows us not only to capture the relevant concepts extracted from the unstructured textual dataset, but also to demonstrate the requirement to map the symptoms of obesity and associated diseases to their higher-level categories to provide a way to standardise the herbal plant concepts.

The results from the three sets of experiments, conducted using WEKA software using Predictive Apriori Algorithm, show higher association for herbs grouped by family name regardless of the dataset. This was achieved by removing small instances of occurrences of certain concepts. However, by removing small instances, there is a risk of losing useful data. The initial set of experiments was based on a restricted number of articles, has demonstrated the need for larger dataset to test our approach. The next set of experiments will involve not only a larger set of textual data but also to investigate fuzzier association rules among our extracted concepts.

References

1. Garvey, W.T., Mechanick, J.I., Brett, E.M., Garber, A.J., Hurley, D.L., Jastreboff, A.M., Nadolsky, K., Pessah-Pollack, R., Plodkowski, R.: American association of clinical endocrinologists and american college of endocrinology comprehensive clinical practice guidelines for medical care of patients with obesity executive summary. Endocrinolog. Pract. **22**, 842–884 (2016)
2. Ahmad, R., Ahmad, N., Naqvi, A.A., Shehzad, A., Al-Ghamdi, M.S.: Role of traditional Islamic and Arabic plants in cancer therapy. J. Tradit. Complement. Med. **7**, 195–204 (2017)
3. Hamid, K.S., Reza, A., Ranjbar, S.H., Esfehani, M.M., Mohammad, K., Larijani, B.: A systematic review of the antioxidant, anti-diabetic, and anti-obesity effects and safety of triphala herbal formulation. J. Med. Plants Res. **7**, 831–844 (2013)

4. Tyler, V.E.: Herbal medicine: from the past to the future. Public Health Nutr. **3**, 447–452 (2000)
5. Pal, S., Shukla, Y.: Herbal medicine: current status and the future. Asian Pac. J. Cancer Prev. **4**, 281–288 (2003)
6. Brand, E., Leon, C., Nesbitt, M., Guo, P., Huang, R., Chen, H., Liang, L., Zhao, Z.: Economic botany collections: a source of material evidence for exploring historical changes in Chinese medicinal materials. J. Ethnopharmacol. **200**, 209–227 (2017)
7. Saad, B., Azaizeh, H., Said, O.: Arab herbal medicines. Bot. Med. Clin. Pract. **16**, 31–39 (2008)
8. Cheng, C.W., Bian, Z.X., Zhu, L.X., Wu, J.C.Y., Sung, J.J.Y.: Efficacy of a Chinese herbal proprietary medicine (hemp seed pill) for functional constipation. Am. J. Gastroenterol. **106**, 120–129 (2011)
9. Afifi, F.U., Abu-Irmaileh, B.: Herbal medicine in Jordan with special emphasis on less commonly used medicinal herbs. J. Ethnopharmacol. **72**, 101–110 (2000)
10. Xie, B., Ding, Q., Han, H., Wu, D.: MiRCancer: a microRNA-cancer association database constructed by text mining on literature. Bioinformatics **29**, 638–644 (2013)
11. He, W., Zha, S., Li, L.: Social media competitive analysis and text mining: A case study in the pizza industry. Int. J. Inf. Manage. **33**, 464–472 (2013)
12. Moreno, A., Redondo, T.: Text analytics: the convergence of big data and artificial intelligence. Int. J. Interact. Multimed. Artif. Intell. **3**, 57 (2016)
13. Tkachenko, M., Simanovsky, A.: Named entity recognition: Exploring features. In: Proceedings of KONVENS, pp. 118–127 (2012)
14. Zhou, X., Peng, Y., Liu, B.: Text mining for traditional Chinese medical knowledge discovery: a survey. J. Biomed. Inform. **43**, 650–660 (2010)
15. Zhu, F., Patumcharoenpol, P., Zhang, C., Yang, Y., Chan, J., Meechai, A., Vongsangnak, W., Shen, B.: Biomedical text mining and its applications in cancer research. J. Biomed. Inform. **46**, 200–211 (2013)
16. Ngo, D.L., Yamamoto, N., Tran, V.A., Nguyen, N.G., Phan, D., Lumbanraja, F.R., Kubo, M., Satou, K.: Application of word embedding to drug repositioning. J. Biomed. Sci. Eng. **09**, 7–16 (2016)
17. Landge, M.A., Rajeswari, K.: A survey on chemical text mining techniques for identifying relationship network between drug disease genes and molecules. Int. J. Comput. Appl. **146**, 5–9 (2016)
18. Chun, H., Kim, J., Tsuruoka, Y., Shiba, R., Nagata, N., Hishiki, T., Tsujii, J.: Automatic recognition of topic-classified relations between prostate cancer and genes from medline abstracts. BMC Bioinform. **24**, S4 (2006)
19. Huang, Y., Wang, L., Wang, S., Cai, F., Zheng, G., Lu, A.: Treatment principles of obesity with chinese herbal medicine: literature analysis by text mining. Engineering **5**(10), 7–11 (2013)
20. Chen, G., Jiang, M., Lv, C., Lu, A.P.: Prediction of therapeutic mechanisms of tripterygium wilfordii in rheumatoid arthritis using text mining and network-based analysis. In: ITME2009 – Proceedings of 2009 IEEE International Symposium on IT in Medicine & Education, pp. 115–119. IEEE, China (2009)
21. Henrotin, Y., Clutterbuck, A.L., Allaway, D., Lodwig, E.M., Harris, P., Mathy-Hartert, M., Shakibaei, M., Mobasheri, A.: Biological actions of curcumin on articular chondrocytes. Osteoarthr. Cartil. Osteoarthr. Res. Soc. Int. **18**, 141–149 (2010)

22. Cai, Y., Wang, G., Yu, X., Zheng, G., Cai, F., Lu, A., Jiang, M.: Basic treatment principles for urinary tract infections with Chinese herbal medicine: an application of text mining. In: 2012 7th International Conference on Computing and Convergence Technology (ICCCT), pp. 1390–1394. IEEE, Korea (2012)
23. May, B.H., Lu, C., Bennet, T.L.: Evaluating the traditional Chinese literature for herbal formulae and individual herbs used for age-related dementia and memory impairment. Biogerontology **13**, 299–312 (2012)
24. Kang, J.H., Yang, D.H., Park, Y.B., Kim, S.B., Korea, I.: A text mining approach to find patterns associated with diseases and herbal materials in oriental medicine. Int. J. Inf. Educ. Technol. **2**, 224–226 (2012)
25. Chen, H.Y., Lin, Y.H., Chen, Y.C.: Identifying Chinese herbal medicine network for treating acne: implications from a nationwide database. J. Ethnopharmacol. **179**, 1–8 (2016)
26. Haruechaiyasak, C., Pailai, J., Viratyosin, W., Kongkachandra, R.: ThaiHerbMiner: a Thai herbal medicine mining and visualizing tool. In: Workshop on Biomedical National Language Processing, pp. 186–187 (2011)
27. Fang, Y.C., Huang, H.C., Chen, H.H., Juan, H.F.: TCMGeneDIT: a database for associated traditional Chinese medicine, gene and disease information using text mining. BMC Complement Altern. Med. **8**, 1–11 (2008)

A Reinforcement Learning-Based Adaptive Learning System

Doaa Shawky(✉) and Ashraf Badawi

Center for Learning Technologies, University of Science and Technology,
Zewail City, Giza, Egypt
{dshawky, abadawi}@zewailcity.edu.eg

Abstract. With the plethora of educational and e-learning systems and the great variation in students' personal and social factors that affect their learning behaviors and outcomes, it has become mandatory for all educational systems to adapt to the variability of these factors for each student. Since there is a large number of factors that need to be taken into consideration, the task is very challenging. In this paper, we present an approach that adapts to the most influential factors in a way that varies from one learner to another, and in different learning settings, including individual and collaborative learning. The approach utilizes reinforcement learning for building an intelligent environment that, not only provides a method for suggesting suitable learning materials, but also provides a methodology for accounting for the continuously-changing students' states and acceptance of technology. We evaluate our system through simulations. The obtained results are promising and show the feasibility of the proposed approach.

Keywords: Adaptive learning · Reinforcement Learning
Computer-supported collaborative learning

1 Introduction

Personalized learning often refers to the individualized instruction and support provided to students, which usually involves the integration of technology in a blended learning scenario [1]. The concept is viewed as the new approach to learning in which "one-size-fits- all" strategy is no longer applicable or acceptable [2]. Personalized learning encompasses several strategies. Usually, student's progress towards a clearly-defined goal is continuously monitored and assessed. In addition, students are provided with personalized learning paths, and they have frequently-updated profiles with weaknesses, strengths, motivation and goals.

In order to provide the aforementioned strategies, we need to build an intelligent learning environment that continuously monitors the variables that affect learning in different settings, and hence, update the suggested learning paths and materials from one learner to another. This is a non-trivial task, since the factors that affect learning can not be modeled or measured in isolation from each other [3–7]. In addition, they are mediated by other factors that may be hidden or unclear. For example, the learning

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 221–231, 2018.
https://doi.org/10.1007/978-3-319-74690-6_22

experience is influenced by learners' affective states, which might not be easily-measured or monitored.

The literature includes several studies that provide promising approaches to personalized learning. For instance in [8], ontology-based models for students, learning objects, and teaching method are proposed. The models consist of four layers that support personalized learning through reasoning and rule-based actions. Also in [9], a personalized learning process is supported by tuning the compatibility level of the learning objects with respect to the learning style of the learner. In addition, the complexity level of the learning objects with respect to the knowledge level of the learner and her interactivity level during the learning process using a modified form of genetic algorithm is modified. Results show the improvement in students' satisfaction. Moreover, in [10], case base planning techniques are used to generate sequences of e-learning routes which are tailored to the students' profiles. Also in [11], a survey on students' modeling approaches for building an automatic tutoring system is presented. The study concluded that for the different modeling tools and methods used, the most common-modeled student's characteristic is the knowledge level and the least common-modeled student's characteristic is her/his meta-cognitive features. However, detecting which set of characteristics is more important is still an open question.

This study develops a framework for personalized learning systems that alleviates some of the shortcomings and challenges to building an effective personalized learning system. The framework is based on the unsupervised machine learning tool; the reinforcement learning (RL). Since personalized learning systems have to be highly-dynamic, RL would be an effective tool for modeling the features of such systems. This is mainly because RL has the potential of dynamically approximating a changing model of the environment. The proposed approach consists of the following steps. Firstly, the learner's state is determined. Secondly, a learning material or path is suggested through a set of actions. Thirdly, based on reinforcement learning, the learner state is updated, in addition, the rewards received by recommended learning paths or material are updated.

The rest of the paper is organized as follows. In Sect. 2, a review on RL is provided. In Sect. 3, the proposed RL-based approach is presented. In addition, in Sect. 4, the system is evaluated and simulation experiments and results are discussed. Finally, Sect. 5 presents the conclusions and outline directions for future work.

2 Reinforcement Learning

Reinforcement learning is inspired by how learning occurs naturally by interacting with the environment, and by how biological systems learn [12]. Similar to all types of learning, it is about mapping situations to actions in order to maximize some rewards. However, the challenge in this type of learning is that, as opposed to other machine learning paradigms, the learner has to discover by herself the best action to be taken in a given situation. Thus, a learning agent must be able to sense the environment and choose the action that would maximize the rewarding function and update her state accordingly. In addition, she has to operate despite the uncertainty about the environment she might have.

As reinforcement learning schemes build environment information through exploration, they are suitable for unsupervised online implementation. A general RL is shown in Fig. 1. The environment can be characterized by the configuration or values of a certain number of its features, which is called its state, denoted at time t as S(t). Each state has a value, dependent upon a certain immediate reward or cost, denoted at time t as R(t), which is generated when it is entered. At each time instance, the agent may take one of a number of possible actions, A(t), which affects the next state of the system, S(t + 1), and therefore the next reward/cost experienced, according to certain transition probabilities. The agent's choice of actions, given the current state of the system, is modified by experience. Thus, an RL system uses its past experience of action taken in a certain system state and reward experienced to update its decision for future actions. A policy of actions to be taken given particular system states is developed over time by the agent as it interacts with the environment.

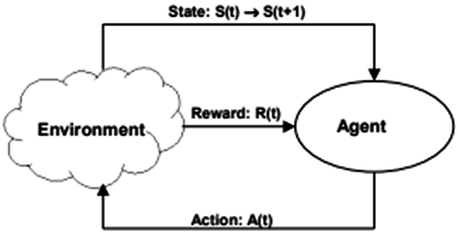


Fig. 1. An RL system [12]

The reinforcement learning problem is usually solved by dynamic programming, Monte Carlo methods, or temporal difference methods (TD) which is a combination of Monte Carlo and dynamic programming [13]. In TD learning, no model is used for mimicking the environment, however the learnt rewards are updated. The main objective is to estimate the value function V_π for a given policy π, which is called the prediction problem. Similar to Monte Carlo methods, TD uses experience to update the estimate of v for the states occurring in that experience. However, in Monte Carlo methods, the updates are done when the return following the visit is known. This is not the case in TD, where the method waits for the next time step $t + 1$ to update the observed reward R_{t+1} and the estimate $V(S_{t+1})$. The simplest TD method is given by (1):

$$V(S_t) \leftarrow V(S_t) + \alpha(R_{t+1} + \gamma V(S_{t+1}) - V(S_t)) \tag{1}$$

Another commonly used method for solving an RL problem is the Q-learning [14]. This algorithm allows learning the optimal policy to accomplish, based on the history of interactions of the system with the environment. In contrast with TD, this algorithm is an off-policy algorithm because no policy is used for suggesting the actions.

The actions are suggested based on some other criterion. This if the system is in state S_i, and it takes the action a_i, it will obtain a reward of r_{i+1}. Each time the system takes an action, given a state, and it receives a reward, an estimation of the scores the state S receives under the action a, denoted by Q(s, a), which is updated based on (2):

$$\mathrm{Q(s,a)} \leftarrow \mathrm{Q(s,a)} + \alpha(\mathrm{r} + \gamma\ \mathrm{maxa}'\mathrm{Q}(\mathrm{s}', \mathrm{a}') - \mathrm{Q(s,a)}) \tag{2}$$

where α is a step rate; r is the observed reward, s' is the new state, $\gamma < 1$ is a discounted factor for the future rewards received under the taken action. $Q(\mathrm{s}', \mathrm{a}')$ is the estimation of the maximum reward that system can measure by taking some future action in the state s'. The complete algorithm is given below.

```
Initialize Q(s, a) arbitrarily
Repeat
  Initialize s
  Repeat
    Choose a from s using $\epsilon$-greedy policy
    Take action a, measure r and observe $s'$
    $Q(s, a) \leftarrow Q(s, a) + \alpha(r + \gamma\ max_{a'}\ Q(s', a') - Q(s, a))$
    $s \leftarrow s'$
```

The Q-learning Algorithm

3 Proposed Approach

This section describes the framework of the proposed system and analyses its main components.

3.1 Main Components

The framework consists of six main components as shown in Fig. 2. The six components are connected to a user interface, and students' database. The system starts by loading the student's static information, in addition to the state-action matrix history. This includes some static data (e.g., gender, major, courses, etc.), in addition to some dynamic data (e.g., state-action-reward history, interactions level, log activities, etc.). Student state is loaded in Step 2. Initially, this represents her state the last time she was logged into the system. If this is her first time, a state that matches her static data is assigned to her. In the third step, an action is suggested, which usually includes a recommended learning material or some engaging material such as some pieces of advice from her instructor in a written, or recorded video or audio, a quotation, or even a joke. In the fourth step, the reward of the taken action is measured. This includes a direct reward where the student is asked to provide a value for her satisfaction level about the recommended material. Moreover, her interactivity level with the system is measured and combined with her satisfaction level to update the reward. Both of these actions are assigned a value out of 5, and these values are used to update the rewards received by the suggested action to be used the next time the student uses the system. In addition, an indirect measure is used which includes the scores of the exams and assignments she received. A negative reward is added to the suggested sequence of actions throughout the semester, if the obtained final grade (in points) is decreased. For example, if the student's previous grade is 2.5, and if the new grade is 2, then this will correspond to –0.5 to be assigned as the final reward received by the set of suggested actions. If this list includes 5 suggested actions, then each one will be assigned a negative value of 0.1, which is the average value. Thus, the main goal of the system is to learn the set of actions for each student's state that will maximize her satisfaction and interactivity with the system during the semester, and at the same time enhances her learning outcomes. In the fifth step, the new state of the student is identified. This is to be done by letting the student choose between the available list of states. She is also asked about proposing a new state to be added to the system if she thinks that none of the provided states can describe her current state. This step is done for the sake of enhancing the performance of the system where the newly-suggested states will be analyzed by an expert and the list of suggested actions for the newly-added states will added to the system. This process is done offline and every while (e.g., at the end of the semester). The main challenge in personalized learning systems using RL is how to determine the State-Action-Reward triplets. In the following subsections, the three main triplets of the proposed RL-based framework will be described in more detail.

3.2 State-Action-Reward

A significant initial stage of constructing a personalized learning system is the selection of appropriate factors that should be considered and represented. The personalization is accomplished efficiently by measuring these factors. In order to determine what factors to be included when designing an effective personalized learning system, a careful and comprehensive investigation of the studies that highlight the factors that affect learning in different settings was performed. Based on this investigation, the factors to be measured can be classified into the following categories.

- Personal Factors
- Social Factors
- Cognitive Factors
- Structural Factors
- Environmental Factors

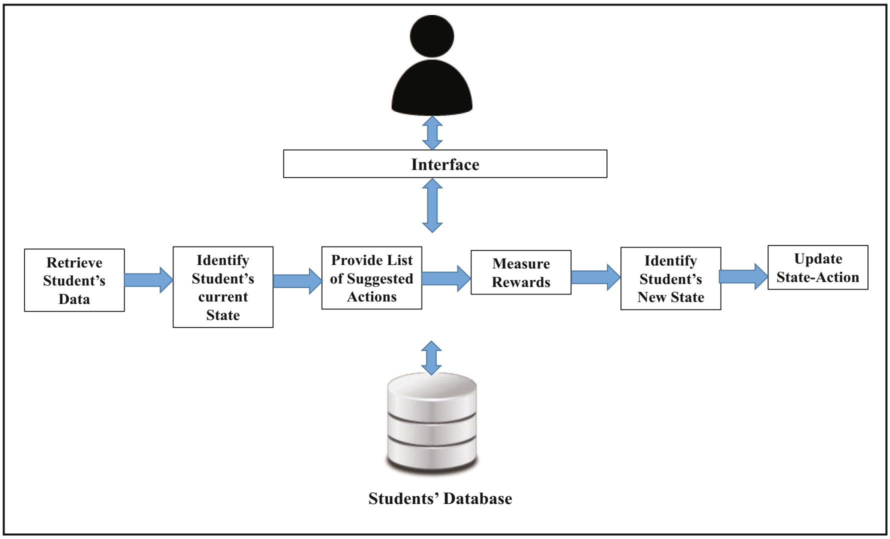

Interface
Retrieve Student's Data
Identify Student's current State
Provide List of Suggested Actions
Measure Rewards
Identify Student's New State
Update State-Action
Students' Database

Fig. 2. The framework of the proposed system

Some of the factors above are individual-level factors and others are group-level factors that should be considered when the learning setting is a collaborative one. Another possible classification for the factors to be considered is as static and dynamic factors. For example, the student's characteristics that are static include email, age, native language. Meanwhile dynamic characteristics are defined and updated each time the student interacts with the system. Some of the factors (the static ones) are set by the student at the beginning of the learning process, while the dynamic ones are usually measured through questionnaires.

Therefore, the challenge is to define the dynamic student's characteristics that constitute the base for the system's adaptation to each individual student's needs.

In the proposed approach, states are represented as a vector X = (x1, x2,…, xn), where *n* is the number of dimensions of each state. Table 1 shows the dimensions of each state with some descriptions on how these states are calculated. In addition, the table indicates a list of suggested initial actions for each possible state. Thus, when the tool is invoked, a vector attached to each learner is populated based on the values measured for each dimension. There is a large number of State-Action pairs, which makes exploring the space of possible actions very expensive. In addition, for lack of space, only a subset of possible state-action pairs are indicated in Table 1. The reward attributed towards each successful action suggestion is measured by two factors; first, the acceptance level as received from the user, second, the long term reward which represents the enhancement in the GPA.

Table 1. State-action pairs

Dimension	Possible levels	Action(s)	How the value is measured
Personality traits	Openness	Stimulate reflective learning styles by linking concepts to real life examples	Questionnaire [15]
	Conscientious	Minimal scaffolding is needed in this case Randomly provide any related learning material	
	Agreeableness	Stimulate conscientiousness by assigning small regular quizzes	
	Neuroticism	Maximal scaffolding Provide enjoyable learning to decrease anxiety Provide ways for organizing information into meaningful units. Remove test anxiety by raising self-esteem and worthiness (e.g., quotes)	
	Extraversion	Chatting and discussion with the "more knowledgeable" colleagues	
Learning styles	Activists	Learning activity need to include projects	Questionnaire [15]
	Reflectors	Explain theory using personal life examples Refer to relevant current events Use hierarchal concepts Provide affordances for summarization	
	Theorists	Ask her to organize the sequence of her thoughts	
	Pragmatists	Provide search tools Provide concept maps Ask her to write algorithms and action plans	
	Auditory	Provide audio or video lessons	

(*continued*)

Table 1. (*continued*)

Dimension	Possible levels	Action(s)	How the value is measured
	Language Visual	Provide graphical illustrations of numbers	
	Language Auditory	Provide oral explanations and numbers Use games and puzzles	
	Numerical Visual	Provide graphical illustrations of numbers	
	Visual-kinesthetic combination	Suggest experiment with self-involvement	
Prior educational achievements	GPA scores in related subjects	More scaffolding is provided for low achievements	Calculated
Intellectual skills	IQ values	More scaffolding for low IQ values	Questionnaire [16]
Perceived satisfaction about the program	5 point likert scale	Provide resources on program's objectives. Highlight and resolve the main reasons for the low satisfaction by top-management	Questionnaire [17]
Motivation	High/low	Motivate peer-peer interactions and communications with those who have high motivation measures	Questionnaire [18]
Social capital	High/low	Provide material that would motivate social presence [19]	By measuring Interactions [19]
Team-related factors: mutually shared cognition, psychological safety, cohesion, potency, and interdependence [20]	High/low	Group students with shared cognitive levels and high cohesion	Responses rates of group members, and interaction between them
Teacher-oriented factors: familiarity with the tool and beliefs	High/low	Provide teachers with instructional guidelines to increase their level of tool acceptance	Questionnaire [21]
Environment-related factors: time poorness, lighting, temperature, noise [22]	Suitable/needs adjustments	Adjust environmental factors to acceptable levels Provide automatic reminders of tasks and assignments deadlines	Sensors or feedback

4 Evaluations

The performance of the proposed framework is evaluated through simulation. In the simulation experiments, a system with 20 states and 20 actions is used. Thus, a state-action matrix is of dimensions 20 × 20. Moreover, the matrix is initially populated with randomly generated Q-values (rewards) that follow the Normal distribution (with mean = 0, and standard deviation = 1). In addition, an ε-greedy approach is used to select the action to be selected with ε set to 0.1. In an ε-greedy policy, actions with maximum rewards are selected with a probability of ε. This allows for exploring the environment, by not necessarily selecting the actions with maximum rewards. The rewards assigned to the 20 available actions for each state are randomly generated. However, for 10 of the available actions, the assigned rewards were negative, while the other 10 actions were assigned positive rewards. The learning rate is set to 0.1, together with other model's parameters. Moreover, the behaviors of 10 students were simulated. A maximum number of 100 iterations were used. Figure 3 shows the number of actions that received positive rewards for each simulated student behavior (denoted by S1 to S10 in the legend) versus the total number of iterations. As shown in the figure, the number of suggested actions that receive positive rewards increases, as the number of iterations increases. This indicates that the simulated system is able to find the best actions to be followed for each student-state pairs after a sufficient number of runs.

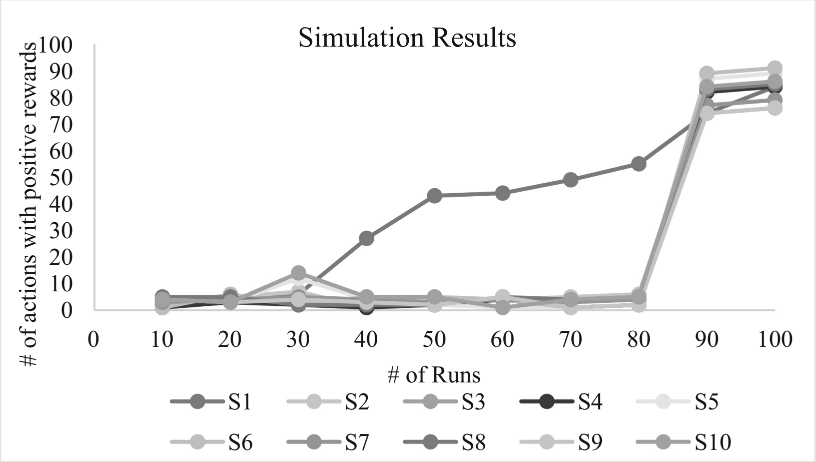


Fig. 3. Simulation results for 20 × 20 state-action matrix for 10 simulated students

5 Conclusions and Future Work

This paper presents a personalized learning framework based on RL. The proposed approach can assist the students to find out what she or he really needs, by investigating the features of a learning material or a sequence that has not been explored before. It also allows for adding the newly-suggested learning sequences by the students and/or the teachers. By investigating the history of state-action-reward for each student, the system will intelligently be able to propose the best learning environment for each student.

As a future work, it is important to add as many actions as possible for each state to allow for the exploration of the optimal one for each student-state pair. The main problem, however, is the large number of possible states or state-action values. This might cause complexity and convergence problems, especially if the model is to be implemented online without the benefit of a repetitive training period. Thus, scalability issues need to be considered in the future work.

References

1. McCarthy, B., et al.: Journey to Personalized Learning (2017)
2. Twyman, J.S.: Competency-Based Education: Supporting Personalized Learning. Connect: Making Learning Personal. Center on Innovations in Learning, Temple University (2014)
3. Shawky, D., Badawi, A., Said, T., Hozayin, R.: Affordances of computer-supported collaborative learning platforms: a systematic review. In: 2014 International Conference on Interactive Collaborative Learning (ICL), pp. 633–651. IEEE, December 2014
4. Fahmy, A., Said, Y., Shawky, D., Badawi, A.: Collaborate-it: a tool for promoting knowledge building in face-to-face collaborative learning. In: 2016 15th International Conference on Information Technology Based Higher Education and Training (ITHET), pp. 1–6. IEEE, September 2016
5. Ashraf, B., Doaa, S.: The need for a paradigm shift in CSCL. In: The Computing Conference 2017. IEEE, London (2017)
6. Said, T., Shawky, D., Badawi, A.: Identifying knowledge-building phases in computer-supported collaborative learning: a review. In: 2015 International Conference on Interactive Collaborative Learning (ICL), pp. 608–614. IEEE (2015)
7. Taraman, S., et al.: Employing Game theory and Multilevel Analysis to Predict the Factors that affect Collaborative Learning Outcomes: An Empirical Study. arXiv preprint arXiv:1610.05075 (2017)
8. Ouf, S., et al.: A proposed paradigm for smart learning environment based on semantic web. Comput. Hum. Behav. **72**, 796–818 (2017)
9. Christudas, B.C.L., Kirubakaran, E., Thangaiah, P.R.J.: An evolutionary approach for personalization of content delivery in e-learning systems based on learner behavior forcing compatibility of learning materials. Telemat. Inform. (2017)
10. Garrido, A., Morales, L., Serina, I.: On the use of case-based planning for e-learning personalization. Expert Syst. Appl. **60**, 1–15 (2016)
11. Chrysafiadi, K., Virvou, M.: Student modeling for personalized education: a review of the literature. In: Advances in Personalized Web-Based Education, pp. 1–24. Springer, Heidelberg (2015)
12. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction, vol. 1. MIT Press, Cambridge (1998)
13. Tsitsiklis, J.N., Van Roy, B.: Analysis of temporal-diffference learning with function approximation. In: Advances in neural information processing systems, pp. 1075–1081 (1997)
14. Watkins, C.J.C.H., Dayan, P.: Q-learning. Mach. Learn. **8**(3–4), 279–292 (1992)

15. Vasileva-Stojanovska, T., et al.: Impact of satisfaction, personality and learning style on educational outcomes in a blended learning environment. Learn. Individ. Differ. **38**, 127–135 (2015)
16. Chamorro-Premuzic, T., Furnham, A.: Personality, intelligence and approaches to learning as predictors of academic performance. Personal. Individ. Differ. **44**(7), 1596–1603 (2008)
17. Eom, S.B., Wen, H.J., Ashill, N.: The determinants of students' perceived learning outcomes and satisfaction in university online education: an empirical investigation. Decis. Sci. J. Innov. Educ. **4**(2), 215–235 (2006)
18. Gagne, R.M.: Learning outcomes and their effects: useful categories of human performance. Am. Psychol. **39**(4), 377 (1984)
19. Dika, S.L., Singh, K.: Applications of social capital in educational literature: a critical synthesis. Rev. Educ. Res. **72**(1), 31–60 (2002)
20. Van den Bossche, P., et al.: Social and cognitive factors driving teamwork in collaborative learning environments: team learning beliefs and behaviors. Small Group Res. **37**(5), 490–521 (2006)
21. Song, Y., Looi, C.-K.: Linking teacher beliefs, practices and student inquiry-based learning in a CSCL environment: a tale of two teachers. Int. J. Comput. Support. Collab. Learn. **7**(1), 129–159 (2012)
22. Clark, H.: Building Education: The Role of the Physical Environment in Enhancing Teaching and Research. Issues in Practice. ERIC, London (2002)

Performance Evaluation of SVM-Based Amazighe Named Entity Recognition

Meryem Talha[1(✉)], Siham Boulaknadel[2], and Driss Aboutajdine[1]

[1] LRIT, Unité Associée Au CNRST, Faculty of Science,
Mohammed V University-Agdal, Rabat, Morocco
meriem.talha@gmail.com, aboutaj@fsr.ac.ma

[2] Royal Institut of Amazighe Culture, Allal El Fassi Avenue, Madinat al Irfane,
Rabat-Instituts, Rabat, Morocco
boulaknadel@ircam.ma

Abstract. A big scope within today's Amazighe society is the amount of information generated each day, mainly within the print and online press where a lot of articles are written daily and need to be processed in some way to appropriately recognize the content within. Indeed, the Named Entity Recognition (NER) has become one of the most fundamental tasks for several natural language processing applications, where texts are analyzed to locate and classify entities into predefined classes. While many algorithms have been proposed for this task, Amazighe NER remains a challenging task and an active research area. In this paper, we managed to achieve an encouraging performance, close to a state-of-the-art NER performance in other languages. The empirical results show that the proposed system achieves more than 80%, regarding the F-measure, when applied to our testing dataset that we have created manually.

Keywords: Named Entity Recognition · Amazighe language · SVM
Learning approach · GATE

1 Introduction

Named Entity Recognition (NER) has been a hot topic in the Natural Language Processing (NLP) community. NER is a critical stage for various NLP applications including machine translation, sentiment analysis, question answering, opinion mining, Event Extraction, and many other application areas. Indeed, NER is one of the key information extraction tasks, it is typically broken down into two main phases: the process of parsing a written text and identifying the textual elements entitled names of entities (NE), such as People (Emmanuel Macron, Donald Trump, Michael Jackson, etc.), Locations (Liberia, Iraq, Morocco, etc.), Organizations (IBM, Orange, Royal Air Maroc, etc.), Numerical expressions and Temporal expressions that are also commonly regarded as names of entities.

There have been many attempts to develop techniques to recognize and classify NEs. They roughly fall into three approaches, that is, hand-crafted or rule-based approach, statistical machine learning-based approach and hybrid approach. Machine learning

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 232–241, 2018.
https://doi.org/10.1007/978-3-319-74690-6_23

based approach has been a fruitful research direction in recent years. It is an effective method and can improve the classification performance of a NER system.

Named Entity Recognition (NER) was first introduced in the Message Understanding Conference-6 (MUC-6) [1]. MUC 6 was solely devoted to English as target language. The MUC 6 tasks have been designed to promote researches in Information Extraction (IE) from unstructured text and NER was considered as an important IE subtask, which basically involves identifying and classifying nominal mentions of persons, organizations and locations, and also numeric expressions of money, percentage and finally temporal expressions of time, dates. The three subtasks correspond to three SGML tag elements: ENAMEX, NUMEX, and TIMEX.

According to the literature, most of researches in the NER field are still extensively being done for English, because it is the most widely spoken language throughout the world. Furthermore, German, Chinese, French, Turkish, Portuguese and Arabic are the languages that lately get good attention and are being actively studied. It can be observed that this task draws attention of the research community at large, and not only of those dealing with the English language, which is well provided with NLP resources and tools. However, work on Amazighe NER has not reached a satisfactory level; there is a limited work on NER systems applied on Amazighe texts.

The aim of this paper is to explore the capabilities of a supervised learning approach which employs SVM as a learning scheme, focusing on the NER task of texts in Amazighe. This task will be to take texts in Amazighe as input and find and classify all entities within them into one of the predefined categories. The categories used, according to the MUC guideline for named entities, are: Person, Location, Organization, Numerical Expressions, Temporal Expressions, Money, and Percentages.

Our contribution aimed also at improving the state of the art of Amazighe NER which is perceived to be comparatively lower than the English NER. In fact, we propose a NER system for the Amazighe language without using any handcrafted features. Also, we have extended the covered named entity types. We are now able to recognize and classify 7 types of named entities; which contain person, locations, organizations, numbers, percentage, money, dates. We aim to make our NER system open source for research purpose, which is believed to be a good contribution to the future development of Amazighe NER in particular and Amazighe language processing research in general.

The remainder of this paper is organized as follows: Sect. 2 begins with a comprehensive overview of the different NER methods, and also touches on how previous work on NER has been approached in different languages in general including the Amazighe language. In Sect. 3, we provide some characteristics of Moroccan Amazighe language, and we draw in Sect. 4 a description about the Amazighe NER Challenges. Section 5 is going into detail about how the experiments are carried out. We describe the proposed system and present the corpus peculiarities, we also give a short description of our classifier and the features used to locate our entities. We continue with discussion about the result obtained and other thoughts that needed to be addressed. Section 6 wraps up the paper with a conclusion. Finally we address what could be done in the future to improve the proposed system.

2 Literature Review

In this section, we review related works from two perspectives; Amazighe named entity recognition and NER in other languages, and we will give a quick overview of the different approaches used for the NER task in general.

2.1 Named Entity Recognition Approaches and Methods

Within the large body of research on NER which have been published in the last two decades, there have been many attempts to develop techniques to recognize NEs. They roughly fall into three approaches, that is, handcrafted rule-based approach, statistical machine learning based approach and hybrid approach.

The rule-based approach relies on handcrafted grammatical rules; these rules are used to locate named entities in a given text using their syntactic and lexical structure with the help of the gazetteers and general dictionary [2].

Some examples of rule-based approach for named entity recognition can be seen in the works of [3], who created the first rule-based system for Turkish language in order to recognize named entities including persons, location, and organization together with time/date and money/percentage expressions. Shaalan and Raza [4] presented a NER system for Arabic using a rule-based approach, which make use of a list of named entities, a set of handcrafted rules and a filtering mechanism. The filter helps mainly to revise the system output by using a blacklist to reject the incorrect named entities.

Differently from the rule-based approach, the machine learning based approach does not require any natural language information. It is based on converting Named Entity Recognition problems to classification problems and then using statistical learning algorithms and some feature representation from a large collection of annotated data to make predictions about named entities in a given texts [2].

Machine learning approaches that have been used for NER are divided into two categories: Supervised (SL) and Unsupervised (UL). The main difference between these categories is that the first one requires the availability of large annotated data in the training stage while the second one does not need an annotated data beforehand, it relies on clustering similar documents or entities together. Indeed, SL cannot achieve a good performance without a large amount of training data, because of data sparseness problem. A hybrid version of these two categories is the Semi-supervised machine learning, which combines annotated and unannotated data for inductive learning. The most popular machine learning algorithms used for NER are Support Vector Machine (SVM), Conditional Random Fields (CRF), Maximum Entropy (ME), and Hidden Markov Models (HMM).

In recent years, different machine learning systems have been proposed for NER tasks, [5, 6] amongst others proposed Hidden Markov Models for finding proper names from text corpora, [7, 8] used Maximum Entropy Models (MEM) and [9] used Conditional Random Fields, for the CoNLL-2003 shared task. Benajiba [10] compared in his thesis the results achieved by "ANERsys" from diverse machine learning (ML) approaches such as the Maximum Entropy, Support Vector Machines, and Conditional Random Fields.

2.2 Previous Amazighe NER Studies

To our best knowledge, the study of Talha et al. [11] is the first study on Amazighe NER. In this research, a rule-based approach has been employed with handcrafted grammars and gazetteers; the system created is able to identify named entities of three different types: person (64%), location (82%) and organization (40%). [12], in contract to the previous system, is able to define five named entity types which include person (83%), location (97%), organization (76%), numbers (95%) and dates (67%). Following this, in [13] a new collection of person, organization, location, numbers and dates gazetteers and sets of rules are constructed as information sources. As a recent study, [14] presented a hybrid NER system for Amazighe, however it didn't attempt a very good improvement of results compared to the rule-based system. In this spirit, we aim to build a machine learning system using SVM and analyze the effectiveness of the proposed method using a new set of features.

3 Amazighe Language

Amazighe Language is one of the oldest languages of humanity. It belongs to the branch of the large Hamito-Semitic linguistic family also called Afro-asiatic. It covers a boundless geographical zone: all of North Africa, the Sahara (Tuareg), and a part of the Egyptian oasis of Siwa. But the nations primarily targeted, by order of demographical significance, are “Morocco, Algeria, Niger and Mali”.

In Morocco, and according to the last census of 2014[1], Amazighe is one of the national and official languages besides the classical Arabic; it is spoken by 27% of the Moroccan population. Otherwise, this language is characterized by the proliferation of varieties due the historical, geographical and sociolinguistic factors. It is spread into three large dialectical areas: Tarifite (4,1%) in North (Rif), Tamazight (7,6%) in Central Morocco (the Mid-Atlas and a part of the High-Atlas) and South-East, and Tachelhite (15%) in the South-West and the High Atlas.

It is so far spoken by 2/3 of Algeria's Amazighe speakers, clustered near Algiers, densely present in Kabyle, which is a thickly populated surface range, and Shawi. Nonetheless, it is also represented by a few groups in the west, east and south of the country. Kabyles are the largest Amazighe group in Algeria.

We should mention that the Amazighe language has not been written - until fairly recently. It is not used widely and certainly not in public contexts such as newspaper, literature, or history. Rather, it is instead employed for private or personal purposes such as letters, diaries, monumental tombs and household decorations. The Amazighe alphabet that was used for these purposes in antiquity is named Tifinagh (which possibly means our invention) and consists of a number of phonetic symbols ultimately related to the Phoenician (Punic) alphabet.

In 2001, thanks to the effort undertaken by IRCAM (the Royal Institute of Amazighe Culture), Amazighe Language has turned into an institutional language nationally recognized in Morocco, and in July 2011, it has become an official language next to the

[1] http://www.hcp.ma/Presentation-des-premiers-resultats-du-RGPH-2014_a1605.html.

classical Arabic. In 2003, Tifinagh IRCAM was established and has been adopted as an official graphic system in Morocco. The Tifinagh IRCAM graphical system has been adjusted and computerized with a specific end goal to give the Amazighe language a sufficient and usable standard writing system. In order to cover all the Moroccan Amazighe varieties, it has a tendency to be phonological.

Tifinagh IRCAM writing system is written horizontally from left to right and contains 33 alphabets (27 consonants; 2 semi-consonants and 4 vowels).

Amazighe is a highly agglutinative language and it makes Amazighe language morphologically rich with very productive inflectional and derivational processes, and it differs from English or other Indo European languages.

4 Amazighe Named Entity Recognition Challenges

In contrast to the significant achievement concerning English or some European languages, the research progress on Amazighe Named Entity Recognition is relatively limited. Applying NER task is very challenging when dealing with Amazighe mainly due to its characteristics. We consider that the major challenges presented by NER in the Amazighe language are the following:

- **No Capitalization:** The absence of the uppercase/lowercase distinction represents a major obstacle for the Amazighe language. Uppercase letters, however, do not occur, neither at the beginning neither within the initials of Amazighe names.
- **Complex Morphological System:** the Amazighe language is agglutinative, meaning it has a rather complex and rich derivational and inflectional morphology.
- **Nested Entities:** The named entities that are considered as nested join two proper names that are nested together to make a new named entity. An example in Amazighe language is "ⵜⵉⵏⵎⵍ ⵏ ⵜⴰⵔⵉⴽ ⴱⵏⵓ ⵣⵢⴰⴷ, tinml n tarik bnu Zyad, Tarik bnou Ziyad School" where "ⵜⴰⵔⵉⴽ ⴱⵏⵓ ⵣⵢⴰⴷ, Tarik bnu Zyad" is the person name and "ⵜⵉⵏⵎⵍ, tinml, School" labels the entire entity as an organization.
- **Entity noun ambiguity:** There are a number of frequently used words (common nouns), which can also be used as names. This happens when a named entity is the homograph of a noun.
- **Lack of standardization and spelling:** The Amazighe language has remained essentially an oral language for a long time. Therefore, the Amazighe text does not respect the standard writing convention.
- **Lack of available Linguistic Resources:** We lead study on the Amazighe language resources and NLP tools (e.g., corpora, gazetteers, POS taggers, etc.). This leads us to conclude that there is a limitation within the number of available Amazighe linguistic resources in comparison with other languages.

5 Experiments and Results

In this section we describe: the proposed method in Amazighe NER, the corpus that we have used for the evaluation, the experiments carried out and the results obtained.

5.1 System Architecture

The Amazighe NER system processes the un-annotated corpus into an NE annotated corpus through a series of processes, i.e. tokenization, sentence splitting, lookup lists, and machine learning classification. The architecture of our proposed system is illustrated in Fig. 1. The following subsections provide the details of each module.

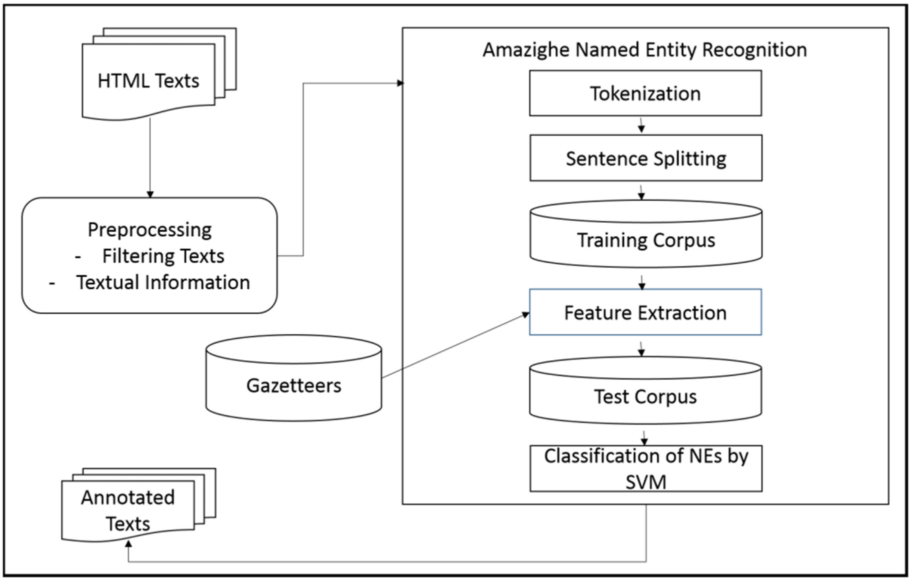


Fig. 1. Workflow of the proposed system.

After converting the html files into plain text format, our system will first go through sentence splitting. The process simply separates the text by sentences. Afterwards, we tokenized our data so that each word is represented as a token. All punctuation characters are considered as a token. The tokens rendered will be forwarded to the Support vector machine module. The module starts the process by recognizing the feature set of every token. Once recognized, the module will annotate the named entities according to their classification. The annotated text will be the output of our system.

5.2 Experiment Data Sets

First level: build Amazighe data. Our Amazighe data set contains more than 900 news articles published online[2], that are collected from a broad range of topics (sports, economics, news on royal activities of His Majesty King Mohammed VI, and many others), containing news that happened over a period of 2 years (dated between May 2013 and July 2015). The articles are selected in such a way that the data set contains

[2] The articles were collected from: http://www.mapamazighe.ma/am/.

different types of information, and that the system's future use will not limited to any particular text type. It consists of nearly 170.000 words, after some data cleaning operations like deleting non-Amazighe words. This data set is manually annotated following the MUC guidelines, ENAMEX (Location, Person, and Organization), NUMEX (Numbers, Percentage and Money) and TIMEX (dates & times) types. During the annotation process, if a named entity is embedded in a longer one, then only the longest mention is annotated. We reserved roughly 800 articles of the data for training and used the remaining for test purposes.

Second level: build Gazetteers. We have prepared approximately 23 different gazetteers that contain different types of named entities such as famous person names (2533 entries), we split them into first name and last name gazetteers in order to identify different combinations. Then, Locations names gazetteer (2318 entries), which includes different location names in Morocco, with names of almost all the countries in the world, cities, states, and geographical names from different sources. And finally names of important organizations (913 entries) such as those of political parties, universities and banks. Four additional gazetteers of date (193 entries), numbers (216 entries), money (14 entries) and percentage (3 entries) have been created.

We have also created manually some lists of trigger words, cue words surrounding the named entities, such as titles like (ⵎⴰⵙⵙ, Mass, Mr) and (ⴱⴰⴱ ⵏ ⵜⴰⵜⵜⵓⵢⵜ ⵜⴰⴳⵍⴷⴰⵏⵜ ⴰⴳⵍⴷⵓⵏ ⵎⵓⵍⴰⵢ, bab n tattuyt tagldant agldun mulay, Sa Majesté le Roi). They have been created by looking at the most frequent left and right-hand-side contexts of the Amazighe NEs. Lists of 468 trigger words were created.

5.3 Support Vector Machine (SVM)

We choose to use the SVM for our task since, according to the literature, it is one of the most successful machine learning methods for NER, and it has achieved a state-of-the-art performance on many NER tasks comparable with those of human taggers.

SVM is primarily a binary classifier, however, when using SVMs for NER, we are confronted with the multi-class problem. The larger the number of classes, the more serious the problem becomes. In this case, extensions to multi-class problems are most often performed by combining several binary machines and using a voting technique to make the final classification decision, in order to produce the final multi-classification results [15]. We can handle the multi-class problem by using a One-Against-All or One-Against-One methods [16].

5.4 Features Used

Feature selection plays a crucial role in the Support Vector Machine (SVM) process. Experiments have been carried out in order to find out the most relevant features for NER in the Amazighe language.

Before all else, each set is preprocessed using the open-source ANNIE system, which is included in GATE. This system generates a number of linguistic (NLP) features. The features include token form, token kind, semantic classes from gazetteer lists and named entity type.

5.5 Experiments Results

The evaluation task for each experiment as described above was done using three different metrics: Precision, Recall and F-measure.

Through experiments our main scope is to investigate the impact of our selected feature for the recognition of ENAMEXs, TIMEX and NUMEX categories using 2 different kernels (Linear & Polynomial). Results are presented in Table 1.

Table 1. Evaluation results.

Named entities	Linear kernel (%)			Polynomial kernel (%)		
	Recall	Precision	F-M	Recall	Precision	F-M
Person	91	72	81	91	73	81
Location	90	74	82	93	78	85
Organization	88	84	86	87	79	83
Number	94	82	88	95	82	88
Date/Time	97	92	94	98	91	94
Money	100	88	94	100	80	89
Percentage	100	100	100	100	100	100

The F-Measure rates of our system on our test dataset are promising since the system is an initial adaptation of a machine learning NER system. The results show clearly that the system varied almost significantly by entity type, and the accuracy scores of the various SVM Kernels within each type are relatively close to each other. Our system performed best in tagging Percentage class with an f-measure of 100%, and worst in tagging Person names with an f-measure of 81%, possibly because of few feature sets used for determining class Person.

For person named entities, our system was able to correctly identify them in almost 80% of the cases. For the location class, our system using the polynomial kernel obtained the best F-measure (85%). While for the organization class the linear SVM bases system achieved good results comparing to the other classes especially while using the linear SVM (86%). Theses scores are quite good. This can be interpreted by the good coverage of our system's person gazetteer and the safe lookup method. However, these scores may seem lower when compared to the other categories (NUMEX & TIMEX). Indeed, according to the literature, most systems tested in other languages have some difficulties with the same category (ENAMEX), it's the most challenging category.

All percentage expressions were efficiently recognized using our systems; this is due to the small frequency of percentage instances in our corpus. While for Money category, our system using the linear SVM achieved the best overall precision (88), more than 8 points better than polynomial kernel results.

Regarding the TIMEX category, precision is lower than recall; however for this type of named entities precision and recall are balanced.

The system was not able to locate correctly all the named entities, some of them were wrongly annotated or missed. This is due to the ambiguity of some Amazighe words (e.g., "ⵎⴰⵙⵙ, Massa" (which is a location entity and also a person title).

A deep analysis of errors obtained has shown that sometimes entities were only partially annotated, particularly with ENAMEX category, this can be illustrated with the following case: in the English entity, "Prime Minister of China Li Keqiang, ⴰⵎⴰⵡⵙⴰⵙ ⴰⵎⵣⵡⴰⵔⵓ ⵏ ⵛⵛⵉⵏⵡⴰ ⵍⵉⴽⵉ ⵜⵛⵢⴰⵏⴳ, amawsas amzwaru n chinwa Liki Tchyang" only Li Keqiang "ⵍⵉⴽⵉ ⵜⵛⵢⴰⵏⴳ, Liki Tchyang" is annotated, which means that the extended length of the named entities and the relatively small size of our gazetteer entries prevented our system from locating all parts of the multiple entities and it may have a negative impact on our results.

Some errors were also caused by the spelling of some entities that were not present in our gazetteer (e.g., location name like Rabat could be written in four different ways as ⵕⴱⴰⵟ, ⵕⴱⴰⵜ, ⵔⴱⴰⵜ, ⵔⴱⴰⵟ, ⵕⵕⴱⴰⵟ, ⵕⵕⴱⴰⵜ).

6 Conclusion and Future Work

This paper reported a SVM-based system for recognizing named entities in Amazighe language. In this research, our main aim is to locate Amazighe named entities then to classify them into seven categories (Person, Location, Organization, Percentage, Money, Number, and Date/Time). One of the most important components in our Amazighe NER system is selecting relevant features. Some experiments have been carried out on moderate sets. Experiments showed that the proposed system is accurate and effective; however, these results can be improved.

As perspectives, we intend to increase the size of our corpus in order to get more relevant results and to have more diversified contents. Besides, we plan to improve the quality of our annotated data. Also, we aim to insert more entries in our gazetteers as much as possible, because it has a major impact on the performance on the NER system. In our future contribution, we will focus on studying and choosing the best features for our NER model. The effects of different features and the interactions among them can be further analyzed and evaluated. In addition, we plan to combine the SVM method with other ones (rule-based or machine learning methods), this combination may be accurate. We are also planning on improving our detection rules and increasing the coverage of the detection rules to cover more named entities combinations, especially for the ENAMEX category.

References

1. Grishman, R., Sundheim, B.: Message understanding conference-6. a brief history. In Coling, vol. 96. pp. 466–471 (1996)
2. Nadeau, D., Sekine, S.: A survey of named entity recognition and classification. Lingvisticae Investig. **30**(1), 3–26 (2007)
3. Dilek, K., Yazici, A.: Rule-based named entity recognition from Turkish texts. In: Proceedings of the International Symposium on Innovations in Intelligent Systems and Applications. pp. 456–460 (2009)

4. Shaalan, K., Raza, H.: NERA: named entity recognition for Arabic. J. Am. Soc. Inform. Sci. Technol. **60**(8), 1652–1663 (2009)
5. Bikel, D.M., Miller, S., Schwartz, R., Weischedel, R.: Nymble: a high-performance learning name-finder. In: Proceedings of the Fifth Conference on Applied Natural Language Processing, pp. 194–201. Association for Computational Linguistics (1997)
6. Zhou, G., Su, J.: Named entity recognition using an HMM-based chunk tagger. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, pp. 473–480. Association for Computational Linguistics (2002)
7. Borthwick, A., Sterling, J., Agichtein, E., Grishman, R.: NYU: description of the MENE named entity system as used in MUC-7. In: Proceedings of the Seventh Message Understanding Conference (MUC-7) (1998)
8. Bender, O., Och, F.J., Ney, H.: Maximum entropy models for named entity recognition. In: Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, vol. 4, pp. 148–151. Association for Computational Linguistics (2003)
9. McCallum, A., Li, W.: Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In: Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, vol. 4, pp. 188–191. Association for Computational Linguistics (2003)
10. Benajiba, Y.: Arabic Named Entity Recognition. Ph.D. thesis, Techninal University of Valencia (2009)
11. Talha, M., Boulaknadel, S., Aboutajdine, D.: NERAM: named entity recognition for amazighe language. In: 21st International Conference of TALN, pp. 517–524. Aix Marseille University, Marseille (2014)
12. Boulaknadel, S., Talha, M., Aboutajdine, D.: Amazighe named entity recognition using a rule based approach. In: 11th ACS/IEEE International Conference on Computer Systems and Applications, Doha, Qatar, pp. 478–484 (2014)
13. Talha, M., Boulaknadel, S., Aboutajdine, D.: L'apport d'une approche symbolique pour le repérage des entités nommées en langue amazighe. In: EGC, Luxembourg, pp. 29–34 (2015)
14. Talha, M., Boulaknadel, S., Aboutajdine, D.: Development of amazighe named entity recognition system using hybrid method. J. Res. Comput. Sci. **90**, 151–161 (2015)
15. Hsu, C.W., Lin, C.J.: A comparison of methods for multiclass support vector machines. IEEE Trans. Neural Netw. **13**(2), 415–425 (2002)
16. Kreßel, U.H.G.: Pairwise classification and support vector machines. In: Advances in Kernel Methods, pp. 255–268. MIT Press (1999)

Trained Neural Networks Ensembles Weight Connections Analysis

Muhammad Atta Othman Ahmed(✉)

Department of Electrical and Electronic Engineering,
University of Cagliari, Piazza D'armi, 09123 Cagliari, Italy
muhammad.ahmed@diee.unica.it

Abstract. Randomization ensemble creation technique well-known as Bagging is widely used to construct trained ensembles of base classifiers. The computational power and demand of Neural Networks (NNs) approved in both researches or in applications. The weight connections of the NNs holds the real ability for the NNs model to efficient performance. This paper aims to analyze the weight connections of the trained ensemble of NNs, as well as investigating their statistical parametric distributions via presenting a framework to estimate the best-fit distribution to the weight connections. As so far the presented work is the first attempt to explore and analyze the weight connections distribution of a trained ensemble of NNs. Obtained results proven that the T-location scale statistical distribution is approximately the best-fit to the weights of the trained NNs ensemble, consequently we aim in our future work to employ the outcomes to withdraw the weight connections value from approximated best-fit distribution instead of training the classifier from scratch.

Keywords: Multiple Classifier Systems · Neural Networks · Bagging
Weight analysis · Parametric statistical distributions · Fitting

1 Introduction

Classification is a part of our real life because we meet simple and complex daily classification problems even in our choices. Over the past years many classification strategies and algorithms have been proposed; considering a single classifier or via combining a set of learners to improve the general classification accuracy [1–3]. Artificial Neural Networks (ANNs) are such a powerful tools in classification [4], function approximation, decision making,... etc. [5]. Several techniques as ANNs training algorithms have been presented to burnish their performance by aiming a better NNs performance in the problem at hand [6–9]. Bellido and Fiesler [10], due to the lack of data and mathematical approaches to describe the inside of NN have to resort the assumption that the weight connection of a trained neural network to be like a Normal distribution. They presented an extensive empirical study of weight distribution in a back-propagation NN

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 242–251, 2018.
https://doi.org/10.1007/978-3-319-74690-6_24

and test formally if the weight of trained NN has indeed a normal distribution. Even they considered a very small invalid probability of rejection of 0.005, the majority of weight distributions investigated were described as NOT Normal. In case of using a simple NN model with no hidden layers, the neural network weight distribution passes the normality test with more than 90% in the case of Gene Promotion dataset. Barbour *et al.* [11], presented a review of theoretical and experimental techniques for analysing the distribution of synaptic weights. Comparing different approaches to analyze the distribution of synaptic weights, Barbour clearly described the obtained distributions from different approaches, as all have a similar shape. They summarise that theoretical analysis through optimality principles show various features of the weight distributions. One of the amazing approaches they highlighted is "Obtaining weight distributions from optimality principles". Considering a Perceptron with 1 binary neuron, we aim the Perceptron to learn N random input-output associations (input patterns) by modifying the weight connections. Considering that the Perceptron has a large number of weight connections $\overrightarrow{W}$, Gardner [12] considered $\overrightarrow{W}$-dimensional space representing all possible configurations of $\overrightarrow{W}$. Only a weight vector subspace of $\overrightarrow{W}$ will satisfy all input-output association. As the number N increases, the subspace of $\overrightarrow{W}$ that satisfy all N input-output associations decreases. They recommend estimating the distribution of weights of NN model below or at maximum capacity. Brunel *et al.* [13] presented an interesting study to compare the Perceptron and purkinje cell from optimal capacity and the weight connections distribution. Summary of their comparison is that below maximum capacity, the non-negative weight distribution has 2 components, about 50% at least are zero weights meanwhile Gaussian distribution found to be fit to the positive connections.

2 Related Work

Machine Learning (ML) computational complexity is still an open interesting area among the researcher of this field. Many studies presented a detailed analysis of distribution-free model learning complexity and cost; presenting their algorithms for learning classes under the uniform distribution. Langly [14] surveyed available approaches to decrease the ML computational complexity via selecting the most relevant features or learning examples. Extreme Learning Machine (ELM) [15] presented to solve the problem of machine learning complexity and computational cost; ELM is a unified learning algorithm with automated feature mapping that can be applied to both classification and regression problems. NVIDIA research [16] proposed 3-steps framework to improve the performance of NNs classifier and reduce it's computational complexity by focusing on learning the weight connections; omitting the unimportant connections and finally re-tune the subset of most relevant weight connections. They successfully reduced the number of connections 13x times keeping the performance accuracy intact. Multiple Classifier Systems (MCSs) is a simple system which provides a promising outcome for most of the machine learning problems; hence it train many different models on same data and consider the average of their predictions [4]. The creation of MCS is computationally expensive, however, because it requires the training of multiple learners. Snapshot Ensembles 2017 [17] recently presented to create ensemble of NN with no additional training cost. They exploit local minima of the error function; Producing N different NN (connection weights) by running the learning algorithm only once, instead of running it for N different times starting from different initial weights. Perthame *et al.* [18] proposed a mathematical framework presenting a new alternative formalism for Training A special class of NN called Spiking NN (SNN). They train 1 SNN on a given input/signal I and stop. After training they consider the obtained weight distribution; corresponding to the used training signal/input W_I; for New J_i; $i = 1 : n$; Inputs/signals/patterns they formalized The convergence of the found weight W_I to the desired W_{J_i} without rerunning the training algorithm. Santucci *et al.* [19] proposed a new approach for defining randomization techniques, inspired by the fact that existing ones can be seen as implicitly inducing a probability distribution on the parameters of a base classifier. Accordingly, that new randomization techniques can be obtained by directly defining a suitable parameter distribution for a given classifier, as a function of the training set at hand. An ensemble can therefore be built by directly sampling the parameter values of its members from such a distribution, without actually manipulating the available training data nor running the learning algorithm. In this way, an ensemble can be obtained even without having access to the training set but having access only to a pre-trained classifier. The constructed ensemble is built using a simulation of bagging. The proposed simulated ensemble achieved a classification performance very close to bagging when NMC used as a base classifier. For the base classifier LDC there is only a partial agreement between the proposed randomization technique and the original bagging. In the case of QDC, the difference in performance was not significant due to the p-value of the statistical test, which is used to compare the proposed randomization approach and the original bagging. Authors clearly highlighted that in the case of non-parametric classifiers such as neural networks, the number of parameters (weight connections) can be very high and at the same time they cannot be related to statistics of the data. Ahmed *et al.* [20] presented the effectiveness and use of various diversity measures to construct ensembles of different base classifiers. Most of the contributions aim to improve, accelerate and make easier the concepts related to machine learning strategies. In this work we produce a detailed analysis of the weight connections of a trained ensemble of 1000 NN created and trained using bagging. The main aim of this work, is to explore the distribution of weight connections aiming to investigate highlight these unexplored informations about the nature of trained NN weight connections. This is the first step towards reaching a learning free ensemble of NN's; via estimating the correct distribution of trained weight connections what leads to not train the ensemble anymore but directly creating it via withdrawing the correct weight values from those found from the distributions.

This paper is organized as follows: In Sects. 1 and 2 we present the preliminaries about related works. In Sect. 3 we present the considered statistical parametric distributions and the proposed algorithm to estimate the best-fit distribution to the weight connections of a trained ensemble of NN. In Sect. 4 we describe the proposed Algorithm 1 and the experiments settings. In the Sect. 5 we present the obtained results and discussion.

3 Weight Connections Distribution Approximation

This section explores a strategy to estimate the best-fit well-known statistical distributions to the weight connection values of a trained ensemble of NNs which created using Bagging. The NNs initializing state has a remarkable influence on the classifier performance. A reasonable choice is to randomly initialize the NNs weights.

3.1 Statistical Parametric Distributions

The proposed method aims to approximate the best-fit well-known parametric distribution to weight connections values of a trained NN ensemble created using bagging. The Algorithm 1 describes the steps of fitting the weight vectors of each neuron over the ensemble of the NN; to a list of 17 continuous and discrete parametric distributions the Table 1 reports the detailed description such as the notations, notions, formula, and parameters of selected statistical parametric distributions.

3.2 Best-fit Distribution Approximation

The test returns **a list of valid distributions** sorted by:

- **NLogL:** Negative of the log likelihood.
- **BIC:** Bayesian information criterion (default).
- **AIC:** Akaike information criterion.
- **AICc:** AIC with a correction for finite sample sizes.

The *Likelihood* $\mathcal{L}(\theta|x)$ of a parameter value, θ; a dats x, is equal to the probability density. Let X be a random variable with a discrete probability distribution p depending on a parameter θ. Then the function:

$$\mathcal{L}(\theta|x) = p_\theta(x) = P_\theta(X = x), \tag{1}$$

considered as a function of θ, is called the *likelihood function* (of θ, given the outcome x of the random variable X). The Negative of Log Likelihood:

$$NLogL = -log\mathcal{L}(\theta|x) = -logp_\theta(x) = -logP_\theta(X = x). \tag{2}$$

Table 1. The considered Parametric Distributions names, formula, notations, and parameters

N	Distribution	Notation	Parameters	Probability density function: $f(x)$
1	Beta	$Beta$	α, β	$\mathrm{x}^{\alpha-1}(1-x)^{\beta-1}\frac{}{B(\alpha,\beta)(1)}$
2	Birnbaum-Saunders	BS	γ, μ, β, ϕ	$\frac{\sqrt{\frac{x-\mu}{\beta}}+\sqrt{\frac{\beta}{x-\mu}}}{2\gamma(x-\mu)}\phi(\frac{\sqrt{\frac{x-\mu}{\beta}}-\sqrt{\frac{\beta}{x-\mu}}}{\gamma})$
3	Exponential	Expon	λ	$\begin{cases} \lambda e^{-\lambda x} & x \geq 0, \\ 0 & x < 0. \end{cases}$
4	Extreme value	Ev	a, b	$e^{-e^{(a-x)/b}}$
5	Gamma	Γ	α, β	$\frac{\beta^{\alpha} x^{\alpha-1} e^{-\beta x}}{\Gamma(\alpha)},$
6	Generalized extreme value	Gev	$s, \xi,$	$\frac{1}{\sigma}\begin{cases} (1+\xi s)^{(-1/\xi)-1}\exp(-(1+\xi s)^{-1/\xi}) & \xi \neq 0 \\ \exp(-s)\exp(-\exp(-s)) & \xi = 0 \end{cases}$
7	Generalized Pareto	Pareto(x_m, α)	α	$\alpha\frac{x_m^{\alpha}}{x^{\alpha+1}}$
8	Gaussian	Gauss	μ, σ^2	$\frac{1}{\sqrt{2\pi\sigma^2}}\ e^{-\frac{(x-\mu)^2}{2\sigma^2}}$
9	Logistic	log	μ, s	$\frac{e^{-\frac{x-\mu}{s}}}{s\left(1+e^{-\frac{x-\mu}{s}}\right)^2} = \frac{1}{4s}\operatorname{sech}^2\left(\frac{x-\mu}{2s}\right).$
10	Log-logistic	llog	α, β	$\frac{(\beta/\alpha)(x/\alpha)^{\beta-1}}{\left(1+(x/\alpha)^{\beta}\right)^2}$
11	Lognormal	logNorm	μ, σ	$\frac{1}{x}\cdot\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(\ln x-\mu)^2}{2\sigma^2}\right)$
12	Nakagami	Nakg	m, Ω	$\frac{2m^m}{\Gamma(m)\Omega^m}x^{2m-1}\exp\left(-\frac{m}{\Omega}x^2\right)$
13	Normal	N	μ, σ	$\frac{1}{\sqrt{2\pi\sigma^2}}\ e^{-\frac{(x-\mu)^2}{2\sigma^2}}$
14	Rayleigh	Rlh	σ	$\frac{x}{\sigma^2}e^{-x^2/(2\sigma^2)},$
15	Rician	Rc	ν, σ	$\frac{x}{\sigma^2}\exp\left(\frac{-(x^2+\nu^2)}{2\sigma^2}\right)I_0\left(\frac{x\nu}{\sigma^2}\right),$
16	t location-scale	tLS	ν, μ, σ	$\frac{\Gamma(\frac{\nu+1}{2})}{\Gamma(\frac{\nu}{2})\sqrt{\pi\nu}\sigma}\left(1+\frac{1}{\nu}\left(\frac{x-\mu}{\sigma}\right)^2\right)^{-\frac{\nu+1}{2}}$
17	Weibull	Wb	λ, k	$\begin{cases} \frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1} e^{-(x/\lambda)^k} & x \geq 0, 0 \\ x < 0, \end{cases}$

Let X be a random variable following an absolutely continuous probability distribution with density function f depending on a parameter. Then the function:

$$\mathcal{L}(\theta|x) = f_\theta(x), \tag{3}$$

considered as a function of θ, is called the *likelihood function* (of θ, given the outcome x of X). The Negative of Log Likelihood (NLogL):

$$NLogL = -log\mathcal{L}(\theta|x) = -logf_\theta(x). \tag{4}$$

For a given a statistical model $\mathcal{M}$ of data x. Let k be the number parameters found in the model. Let $\hat{\mathcal{L}}$ be likelihood Maximum function for the model; i.e. $\hat{\mathcal{L}} = P(x|\hat{\theta}, \mathcal{M})$ are the parameter values that maximize the likelihood function. The Akaike information criterion is defined in Eq. 5:

$$AIC = 2k - 2\ln(\hat{\mathcal{L}}). \tag{5}$$

The Akaike information with a correction for finite sample sizes (AICc) is defined in Eq. 6:

$$AICc = -2\mathcal{L}(\theta|x) + 2k(k+1)/(n-k-1). \tag{6}$$

AIC and AICc can be used as a criteria to compare different models for the same data to estimate the best-fit model. The model with the smallest value, as discussed in Akaike [21], is usually the preferred model. The BIC presented in [22]; the corrected Akaike's Information Criterion (AICc) and the Bayesian Information Criterion (BIC) are information-based criteria that assess model fit. Both are based on $NLogL$. The Bayesian Information Criterion (BIC) is defined in Eq. 7:

$$BIC = -2\mathcal{L}(\theta|x) + k\ \ln(n). \tag{7}$$

When comparing the BIC values for two models, the model with the smaller BIC value is considered better. AIC can be derived from the BIC approximation to the Bayes factor. Due to the model selection literature, it is wrong to consider that AIC and BIC selection are directly comparable as if they had the same objective target model, but they are not.

4 Weight Connections Best-fit Distribution Estimation

In this section the proposed algorithm is defined and the performed experiments is described. The experiments were performed on a 39 data sets; 6 artificial data sets generated as 2 Gaussians class problem and well-known data sets obtained from the UCI machine learning repository. Each data set is divided into a training set of size 0.4, a validation set of size 0.3 and the remaining instances assigned to the test set. We trained a 1000 NN classifier on 1000 bootstrap replica of the training set. A validation check is one of the stop criteria for training the NN classifier. The scored single classifier and the ensemble accuracy on the test found to be reasonable. The Algorithm 1 describes the approach to approximate the best-fit statistical parametric distribution to each given input weight vector $\overrightarrow{W}_i$ of the trained NN ensemble. Highlighting a selected example of our experiments performed on the well-known two class problems named **Breast Cancer** dataset, using only the first two features due to the simplicity of the used neural network structure; in order to restrict our investigation on a fixed small number of weight connections. Starting with a simple structure multi-layer feedforward NN with 2 neurons in the input layer (only 2 features per instance used) 3 neurons in the hidden layer and 1 neuron in the output layer; considering the weight connections of 9 neurons per each NN classifier. Using bagging ensemble creation method, we create an ensemble E of size $E = 1000$ NN classifier. The weight connection vectors of each neuron of the trained 1000 NN classifiers; $\overrightarrow{W}_i = \{w_{iC_1}, w_{iC_2}, \ldots, w_{iC_N}\}$ where N = 1000 is the number of ensemble base classifiers; where i is the number of neurons per NN classifier $i = [1, 2, \ldots, 9]$.

Algorithm 1. Describes the proposed best-fit valid parametric distribution estimation technique.

```
Require: A data $\overrightarrow{X}$.
Ensure: Best-Fit parametric probability distribution to data X.
  For Every Distribution in Table 1; do
      Compute the distribution parameters ($\mu$, $\sigma$,... etc.).
      Compute the NLogL, AIC, AICc, BIC.
  End for
  Sort Distributions ascending according to each of NLogL, AIC, AICc, BIC.
  Return Best-fit Distribution with NLogL, AIC, AICc, BIC value.
```

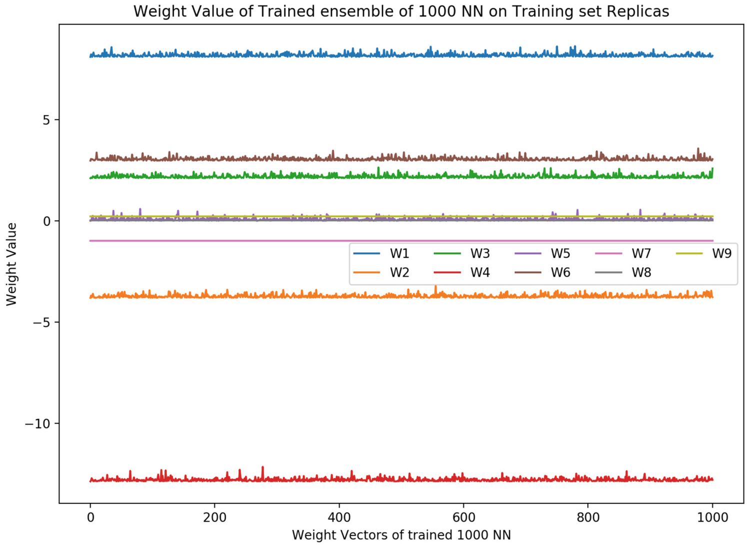


Fig. 1. Weight vectors of trained 1000 NN classifier using Bootstrap sampling; an example of results analysis on Breast Cancer dataset.

5 Results and Discussion

The single neuron weight values is analyzed over the trained 1000 classifier created using bagging. Applying Algorithm 1 using NLogL, AIC, AICc,BIC Eqs. 4, 5, 6, 7 as sort index to select the best-fit distribution, on the weight connections of a trained ensemble using the data set breast cancer; Fig. 1 shows the weight vectors of 9 neurons. The summary of experiments analysis is shown in Fig. 2; it shows the best-fit distributions histogram for each dataset; considering 9 neurons per NN classifier, each corresponding neuron weight vector $\overrightarrow{W}_i$, $i = [1, 2, \ldots, 9]$ that the t-Location Scale distribution was found to be the best fit parametric distribution to 0.75% of the total of 351 weight connections. Considering the silent weight connections (W_7, W_9) in the output layer as shown in Fig. 1. The most relevant conclusion of Fig. 1; that 7/9 weight connections

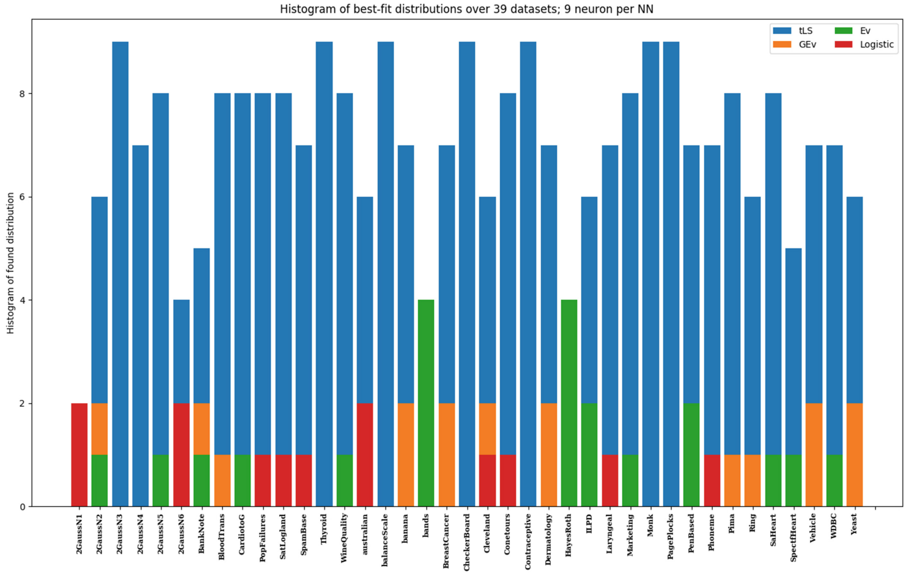


Fig. 2. Histogram of found best-fit parametric distributions to the weight connections of trained ensemble of NN per 39 dataset.

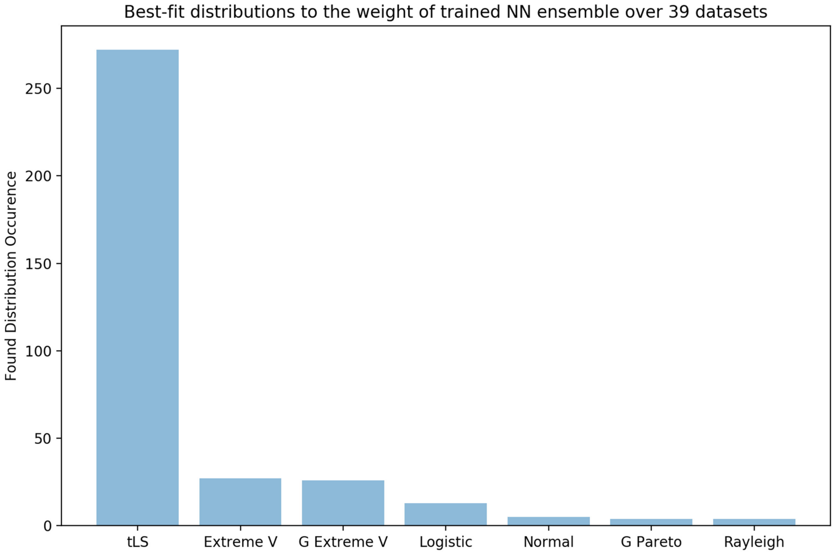


Fig. 3. Summary of the histogram of best-fit parametric distributions over a 39 datasets; 9 neurons per classifier; the total of 351 weight connections; the t-LS distribution fits to 75% of total relevant weight vectors.

found to be-fit to the t-Location Scale meanwhile the other 2/9 silent weight connections fits to the G-Extreme Value. We recall the summary of [16]; reaching that this 2 silent connections are candidates to be omitted and ignored. The Algorithm 1 results confirm that most weight connections of the neural network used are approximately fit the t-Location Scale distribution, the histogram of approximates best-fit distributions shown in Fig. 3.

6 Conclusion and Future Scope

The weight distribution of a trained NN's ensemble created using Bagging is analyzed using the proposed framework presented in Sects. 3.1 and 4. The used approach to estimate the best-fit parametric distribution to the weight value of a single neuron of each NN classifier in the ensemble uses maximum likelihood to compute the parameters of the attempt to fit distribution. Considering the weight distribution of each neuron in the NN classifier for each dataset, this is our future scope to use the estimated best-fit distribution to not train the classifier on new bootstrap replicas of the training set, but automatically assign the weight value of the neuron from already estimated as a best-fit weight distribution. Considering to the optimal principle to investigate the weight distribution highlighted by some references, we aim to re-investigate the weight distribution of a trained ensemble of neural networks created using bagging when it raised the number of instances in the training set until the classifier becomes over on reaching its maximum capacity, that aimed to produce more details and reliable analysis. This is the first step towards our main approach "machine-free learning", targeting the elimination or the maximum reduction of the learning computational cost.

References

1. Breiman, L.: Bagging predictors. Mach. Learn. **24**(2), 123–140 (1996)
2. Rogova, G.: Combining the results of several neural network classifiers. Neural Netw. **7**(5), 777–781 (1994)
3. Giacinto, G., Roli, F., Fumera, G.: Design of effective multiple classifier systems by clustering of classifiers. In: Proceedings of 15th International Conference on Pattern Recognition, vol. 2, pp. 160–163. IEEE (2000)
4. Kuncheva, L.I.: Combining Pattern Classifiers: Methods and Algorithms. Wiley, Hoboken (2014)
5. Huang, G.-B., Saratchandran, P., Sundararajan, N.: A generalized growing and pruning RBF (GGAP-RBF) neural network for function approximation. IEEE Trans. Neural Netw. **16**(1), 57–67 (2005)
6. Shi, B., Bai, X., Yao, C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE Trans. Pattern Anal. Mach. Intell. **39**(11), 2298–2304 (2017)
7. El-Sayed, M.A., Khafagy, M.A.: An identification system using eye detection based on wavelets and neural networks. arXiv preprint arXiv:1401.5108 (2014)

8. Wu, J., Bai, X., Loog, M., Roli, F., Zhou, Z.-H.: Multi-instance learning in pattern recognition and vision (2017)
9. Zhou, Z.-H., Jianxin, W., Tang, W.: Ensembling neural networks: many could be better than all. Artif. Intell. **137**(1), 239–263 (2002)
10. Bellido, I., Fiesler, E.: Do backpropagation trained neural networks have normal weight distributions? pp. 772–775. Springer, London (1993)
11. Barbour, B., Brunel, N., Hakim, V., Nadal, J.-P.: What can we learn from synaptic weight distributions? Trends Neurosci. **30**(12), 622–629 (2007)
12. Gardner, E.: The space of interactions in neural network models. J. Phys. A Math. Gen. **21**(1), 257 (1988)
13. Brunel, N., Hakim, V., Isope, P., Nadal, J.-P., Barbour, B.: Optimal information storage and the distribution of synaptic weights. Neuron **43**(5), 745–757 (2004)
14. Langley, P., et al.: Selection of relevant features in machine learning. In: Proceedings of the AAAI Fall Symposium on Relevance, vol. 184, pp. 245–271 (1994)
15. Huang, G.B., Zhou, H., Ding, X., Zhang, R.: Extreme learning machine for regression and multiclass classification. IEEE Trans. Syst. Man Cybern. Part B (Cybernetics) **42**(2), 513–529 (2012)
16. Han, S., Pool, J., Tran, J., Dally, W.: Learning both weights and connections for efficient neural network. In: Proceedings of Advances in Neural Information Processing Systems, pp. 1135–1143 (2015)
17. Huang, G., Li, Y., Pleiss, G., Liu, Z., Hopcroft, J.E., Weinberger, K.Q.: Snapshot ensembles: train 1, get m for free. arXiv preprint arXiv:1704.00109 (2017)
18. Perthame, B., Salort, D., Wainrib, G.: Distributed synaptic weights in a LIF neural network and learning rules. Phys. D Nonlinear Phenom. (2017)
19. Santucci, E., Didaci, L., Fumera, G., Roli, F.: A parameter randomization approach for constructing classifier ensembles. Pattern Recogn. **69**, 1–13 (2017)
20. Ahmed, M.A.O., Didaci, L., Fumera, G., Roli, F.: An empirical investigation on the use of diversity for creation of classifier ensembles. In: International Workshop on Multiple Classifier Systems, pp. 206–219. Springer (2015)
21. Akaike, H.: A new look at the statistical model identification. IEEE Trans. Autom. Control **19**(6), 716–723 (1974)
22. Schwarz, G., et al.: Estimating the dimension of a model. Ann. Stat. **6**(2), 461–464 (1978)

An Empirical Analysis of User Behavior for P2P IPTV Workloads

Mohamed Elhoseny[1], Abdulaziz Shehab[1](✉), and Lobna Osman[2]

[1] Faculty of Computers and Information,
Mansoura University, Mansoura, Egypt
{mohamed_elhoseny,abdulaziz_shehab}@mans.edu.eg

[2] Department of Electronics and Communications Engineering,
Delta Higher Institute for Engineering & Technology, Mansoura, Egypt
englobna20@yahoo.com

Abstract. The interest for video delivery systems over the Internet has been gradually growing up last years. It has already become a major application due to clients' interest of video content and persistent development of network technologies. Users' behavior is playing an increasingly crucial role in the performance of such applications. This paper proposes an efficient analysis of the user's behavior for a long-running P2P IPTV service infrastructure developed and maintained by Lancaster University. The proposed analysis presents remarkable parameters that could be helpful for the service provider to consider for their network design.

Keywords: P2P · Watching behavior · User behavior analysis · Playback
Video duration

1 Introduction

Multimedia communication systems can improve the level of human-computer interaction (HCI) by providing audio, video, and other media along with traditional media such as text, graphics, and images. Over the past decade, the role of the Internet-based video delivery has grown significantly. Video delivery requires high bandwidth that is a significant obstacle in the network infrastructure. Though recent networks have improved bandwidth availability, network interruptions are still an issue for high-quality video delivery [1]. Nowadays, one of the most popular Internet applications is Video on Demand (VoD) which has recently attracted more and more clients over the Internet. In 2014, more than 1 billion unique users visited YouTube each month, over 6 billion hours of video are watched each month, and 100 h of video are uploaded to YouTube every minute [2].

Many years ago, traditional VoD systems rely on client-server architecture where videos are only stored in centralized media servers. The major problem was that the bandwidth of these media servers often turned to be a bottleneck for the whole system. In contrary, each peer signed in a video delivery session in P2P video streaming become a contributor to other neighboring peers [3]. Inside P2P networks [22, 23], users'

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 252–263, 2018.
https://doi.org/10.1007/978-3-319-74690-6_25

behavior in watching sessions could help improving the overall performance of video delivery process, such as a higher peer's departure is significantly affected other peers [4]. Researchers have applied various different paradigms, but certain issues like the playout latency and reliability of service have never been near optimal. Recent measurement studies [4–7] have presented new findings and problems about current day P2P media streaming systems.

When the service provider provides VoD service, each video title has different popularity and the popularity of a video correlates with the request rate for the video. The popularity of a video may decrease with time since user interest for a video decreases after watching it. It may be also changed due to external factors, such as appearance of new videos and recommendation of videos. Users may have different preferences for each video and have diverse preferable time to watch a video. However, the overall request patterns from all clients follow a uniform curve and constitute daily and weekly access patterns.

The reader can get some interesting related information and publications in the literature. Parameters such as session length, video type, video duration, and user's interactions, etc. have been analyzed in [8, 9]. User's connection, user's origin, and quality requested are studied in [10]. The quality of service (QoS) evaluated in [10, 11] revealed different parameters, such as packet loss, jitter, and quality that was the focus of their proposal. On the other hand, another parameter widely analyzed is content popularity [8, 9, 12, 13]. The study of such a parameter is mainly works at distribution of users' reproductions. Acharya et al. [14] proposed a user accesses to video objects on the web. Their conclusion showed that video popularity did not follow a certain distribution. Other studies and tools have appeared based on some of the results in these papers. These studies were able to simulate real service behavior and carry out performance evaluations like [15]. In their proposal presents a tool for VoD service analysis using simulated workload. Moreover, Veloso et al. [16] proposed an analysis of live video streaming workloads from a video server utilized by one of the popular content providers in Brazil. Sripanidkulchai et al. [17] proposed an analysis of live streaming workload characterizing video popularity, session duration and transport protocol used.

In this paper, we have analyzed the user behavior of one of the P2P VoD services. User behavior was analyzed from the real Lancaster Living Lab, which supports high quality live and on-demand content distribution service. The proposed analysis is very important parameters for the service provider to consider for their network design. The rest of the paper is organized as follows: Sect. 2 presents a description for the proposed tracker's recommender agent. Section 2 presents the dataset that is used as a case study. The content description of the data set is described in Sect. 3. In Sect. 4, the user behavior analysis is reported. Finally, conclusion and future work are drawn in Sect. 5.

2 Tracker's Recommender Agent

As a fast growth in population of P2P VoD system, users' behavior is playing an increasingly crucial role in the performance of video system. Therefore, user watching and sharing behavior become much more important for system performance than ever before. User behavior analysis will be presented, in detail, in Sect. 4.

A good streaming protocol should accommodate a variety of users' watching habits. In general, movies are ranked based on the number of people who watch it. With the help of Tracker's Recommender Agent, we can do more than just counting the number of people. For example, we can tell how long a given movie is watched on an average. We can also tell whether people tend to watch a movie smoothly or jumping from one scene to another. The users' watching behavior is taken into account because it is not only a concern for watchers, but also because watching behaviors affect the media sharing in the VoD environment. The latter is necessary for VoD system design and is different from almost known P2P video delivery systems. The P2P-based VoD systems involve data-sharing among peers. The efficiency of sharing is related to the topology organization of the network, the neighbor selection, and the scheduling mechanism. However, the efficiency is almost irrelevant to users' behavior analysis. In this paper, unlike other systems, the user's watching behaviors affect the sharing efficiency of a VoD system. Thus, it is possible to improve the sharing efficiency of VoD system through proper (even adaptive) content designs and scheduling. After a user selects a video and starts watching it, the user may find the video uninteresting and the user may employ interactive VCR functions such as Fast Search (FS) before the streaming for the movie is finished.

User session can be defined as the time period when a sequence of interactive actions is requested from the same user on a set of videos. Possible user interactions in a session can be classified as follows [18, 21].

- **Play/Resume:** start video playback from the beginning or start playback from other location.
- **Pause:** stop playback with picture.
- **Jump Forward (JF)/Jump Backward (JB):** move to a specific location without picture and sound in forward or backward direction.
- **Fast Search (FS)/Reverse Search (RS):** move to a specific location with picture and sound in forward or backward direction.
- **Slow Motion (SM):** playback slowly to forward direction with picture and sound.
- **Stop/Abort:** stop playback and ending the connection.

Users access the service during a session, which is composed by one or more movies with in-between periods called think time between them. There are two essential elements in a session that must be determined: the number of movies watched and the think time. Each movie begins with a play interaction and continues to the end of the audio/video or to a stop interaction. During this time users can perform intermediate interactions (pause, forward and backward jumps, play, stop), generating paused periods, when no information is being delivered, and active periods, when users are receiving data.

Figure 1 shows the conceptual model of the user behavior in a session, taking into account if the request for a video is success or fail, categorizing the characterization for short, medium, and long videos, and taking into account all the available user interactions (pause, jump, play, and stop). At the end, the user can request a new movie or end the session. Due to the variability in the lengths of the offered videos in the studied dataset, unlike [19], we have differentiated the characterization for short (less than 10 min), medium (10–20 min) and long videos (greater than 20 min).

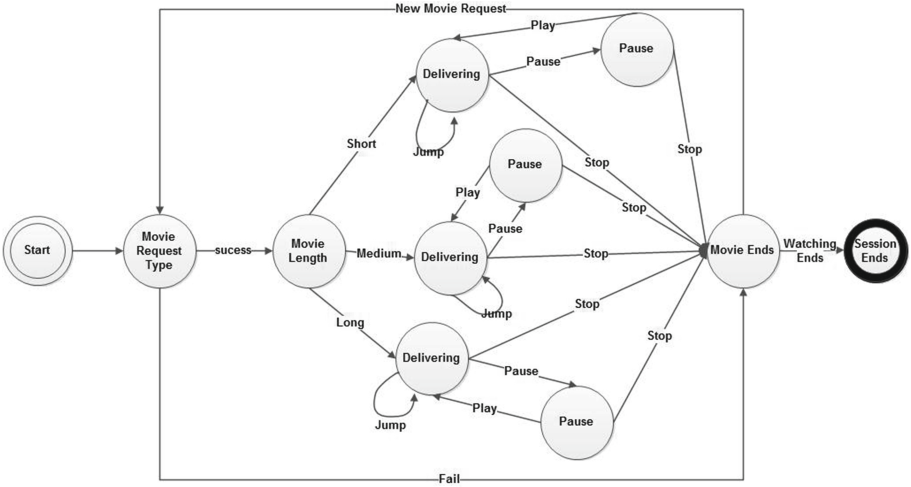


Fig. 1. Conceptual model for user behavior

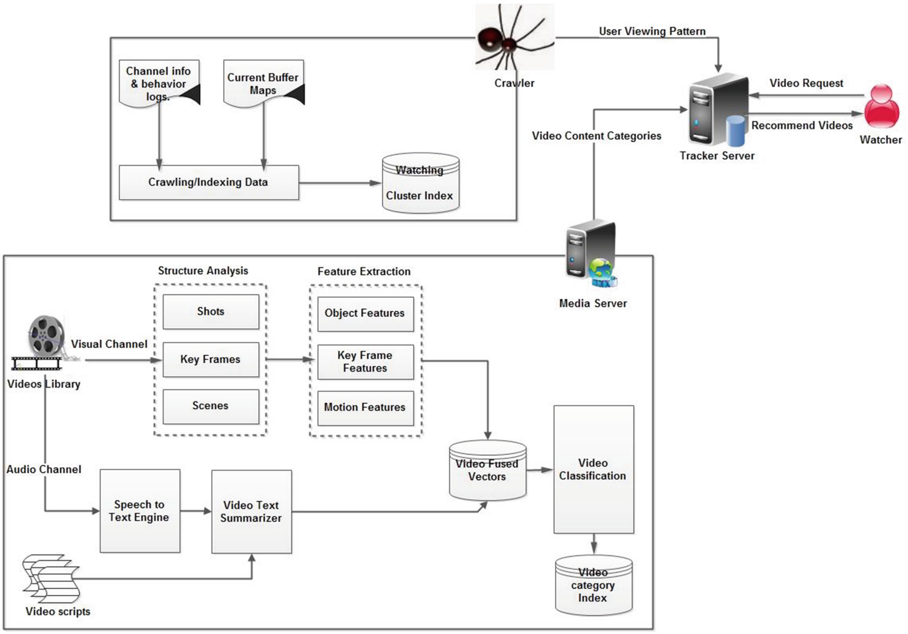


Fig. 2. The case study

As shown in Fig. 2, Tracker's Recommender Agent could be divided into two primary units: (1) crawler unit which runs online during getting the service and (2) video indexing unit which runs in offline mode at the media server side.

2.1 Crawler Unit

Through analysis of large volume of user behavior logs during playing multimedia streaming, the crawler could extract a user viewing pattern. At the first phase, by tracing a VoD client, the crawler capture the interactive packets between the local VoD peer and others. At the second phase, the traced data were fed into a dumping tool that can filter data into a text file with composed conditions, such as source IP/port, destination IP/port, and streaming protocol type. When the crawler is running, it first reads a channel's file, and then starts crawling data from other peers on each channel. By using these phases, the crawler will be able to observe (a) Which users play a video successively and passively from the beginning? (b) Which users rarely view a movie till the end? (c) Which users play the video by jumping? i.e., they watch some scenes while skipping some others. Crawler characterize peers' watching behavior through creating a watching index of data, defined as the maximum chunk ID that a peer has in his/her sharing buffer, and chunk's sharing profile, defined as the number of online copies of a given chunk.

2.2 Video Indexing Unit

Most of video indexing systems rely only on visual video content ignoring the audio content. Many videos may be un-similar in visual content but they are correlated in the main subject particularly in the educational contents. According to the proposed recommender, the videos are passed through a splitter generating visual content channel and audio content channel. In the visual channel, generally, videos are organized according to a descending hierarchy of video clips, scenes, shots, and frames. The media server perform the process of Video Indexing and then notify the tracker with different video categories. This unit contains three main steps are followed: (1) Video structure analysis, (2) feature extraction, and (3) video classification. Video structure analysis aims at segmenting a video into a number of structural elements that have semantic contents, including shot boundary detection, key frame extraction, and scene segmentation. Feature extraction includes static features in key frames, object features, and motion features. The task of video classification is to find rules or knowledge from videos using extracted features and then assign the videos into predefined categories. The multimedia service contents have been classified into various subsections according to their subject. In the audio channel, the voice are first passed to speech to text engine, then the result go through video text summarizer, and finally text similarity measure are performed. If the video scripts are available, the speech to text process could be ignored.

User behavior was analyzed from the real Lancaster Living Lab. The presented study has been performed on NextSharePC dataset [20] which runs on a long-running P2P IPTV service infrastructure developed and maintained by Lancaster University. The service supports high quality live and on-demand content distribution which covers personal and mobile devices. Content services on campus are reachable via Ethernet

(100 Mbit/s) or 802.11 g WiFi (54 Mbit/s). The dataset is covering the period of October 2011–April 2012 with a total of six months.

To capture user activities, three time-coded events of media playback (media play request, media play started, and media play stop) are reported by all end devices to the statistics service. The media play request event records the timestamp and media information of user's request, which is usually triggered by a click on the icon of a live channel or VoD item. The media play started event is defined as the time that the first video frame is rendered for display. When playback is stopped, either explicitly by the stop button or implicitly by the media play request event for another content, the media play stop event is registered.

The timestamp of media play request, media play started, and media play stop enable a number of comprehensive analysis, such as the video loading time (the time difference between media play request and media play started), viewing time (the time difference between media play started and media play stop), and user behavior and program popularity (combining playback events with the program title).

3 Content Description

The multimedia service contents have many subsections according to their subject. Some of these subjects are News, Music, Tourism, Science, Cinema, Comedy, Cartoons, Sports, and Others. Table 1 summarizes the overall statistics of the studied dataset.

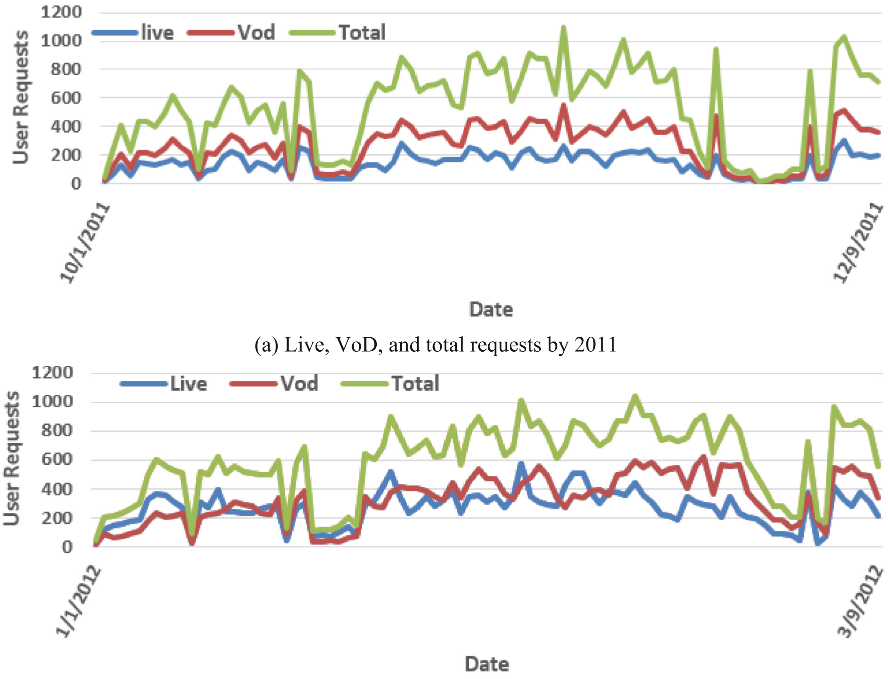
Table 1. Data set summarized statistics

Duration	6 months	Min video duration	0:02:00
Total sessions	78678	Max video duration	13:28:00
Total channels	21	Total videos	3096
Unique users	63096		

4 Results of Peers' Watching Behavior

Figure 3 shows the daily live, on-demand, and total requests for video contents of the first (Oct–Dec 2011) and second (Jan–Mar 2012) time intervals. We notice that the VoD becoming increasingly popular while the requests for live remaining relatively steady. By December, more than one out of every two viewing requests in the entire service were attributed to the on-demand service. By March 2012, the number of VoD requests is more than double that of live requests. The day of 3/11/2012 registers the maximum number of daily requests with 447 live and 594 on-demand requests.

Figures 4 illustrates the channel and the video popularity over the live and VoD. For channel popularity, we observe that BBC One is the most popular channel, followed by BBC Three, E4, and BBC Two. Due to low frequency of many videos, they are grouped together in a cluster named others that represent 53.34% of total live and on-demand videos. For video popularity, in both live and on-demand videos, "Family Guy" is the most watched video, a total of 5875 (2042 in live and 3833 in VoD) requests are received

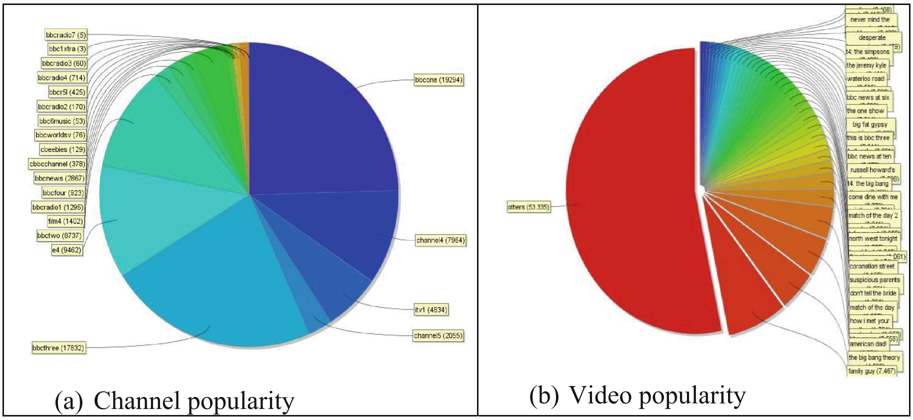


(a) Live, VoD, and total requests by 2011

(b) Live, VoD, and total requests by 2012

Fig. 3. Daily video requests

once created. The most favorite videos are "the family guy" (7.46%) followed by "the big bang theory" (4.60%), American Dad! (4.27%), and BBC news (3.56%) in a descending order.



(a) Channel popularity

(b) Video popularity

Fig. 4. Channel and video popularity for both live and VoD sessions

In order to group videos by their lengths, the media durations are discretized into three clusters short (<10 min), medium (10–20 min), and long (>20 min). The results shown in Fig. 5(a) states that 94% are long, 4% are medium, and 2% are short of the available videos.

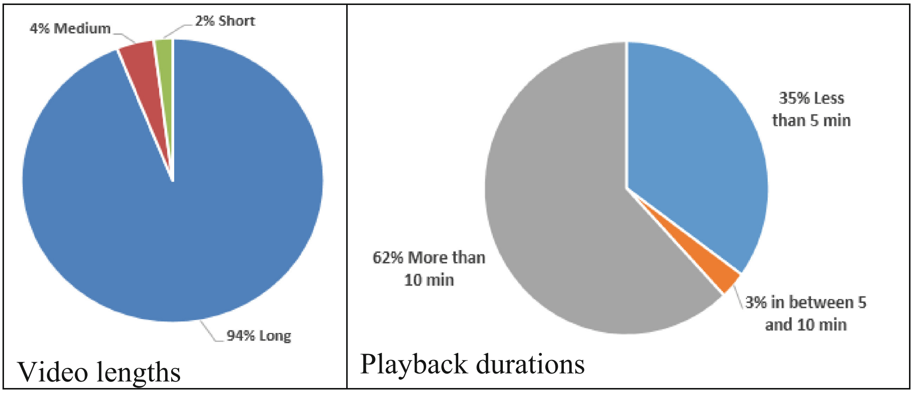


Fig. 5. Pie chart for video lengths categories and playback durations

To characterize how long a video is watched, we use two metrics: video playback time and watching ratio. Video watching time is a measure reflecting the absolute time that a video is watched. By subtracting start time and stop time, the playback duration could be estimated. In reality, however, different videos have quite different lengths. To adapt to the significant variance in video length and obtain a general representation of video watching time, we also use watching ratio as a measure for quantifying the percentage that a video is watched. Watching ratio can be evaluated by calculating the playback to video length ratio. For a video k with a duration of T and playback time P(k), its watching ratio, denoted by W(k), is calculated as: W(k) = P(k)/T.

By analyzing the playback time, as shown in Fig. 5(b), we found that 35% of the users remains watching in sessions for less than 5 min. A small slice, 3%, stays for a period between 5 and 10 min where the majority of users, 62%, remains for more than 10 min.

To characterize how long a video is watched, we use two metrics: video playback time and watching ratio. Video watching time is a measure reflecting the absolute time that a video is watched. By subtracting start time and stop time, the playback duration could be estimated. In reality, however, different videos generally have quite different lengths. To adapt to the large variance in video length and obtain a general representation of video watching time, we also use watching ratio as a measure for quantifying the percentage that a video is watched. Watching ratio can be evaluated through calculating the playback to video length ratio. That is, for a video with duration T and playback time P(T), its watching ratio, denoted by W(T), is calculated as: W(T) = P(T)/T. By analyzing the playback time, as shown in Fig. 5(b), we found 35% of the users remains watching in sessions for less than 5 min. A small slice, 3%, stays for a period in between 5 and 10 min where the majority of users, 62%, stays for more than 10 min.

Figure 6 shows the total number of requests that was occurred by each user. The maximum number of requests records 20 request. In general, the majority of users' request count range from 5 to 20 requests.

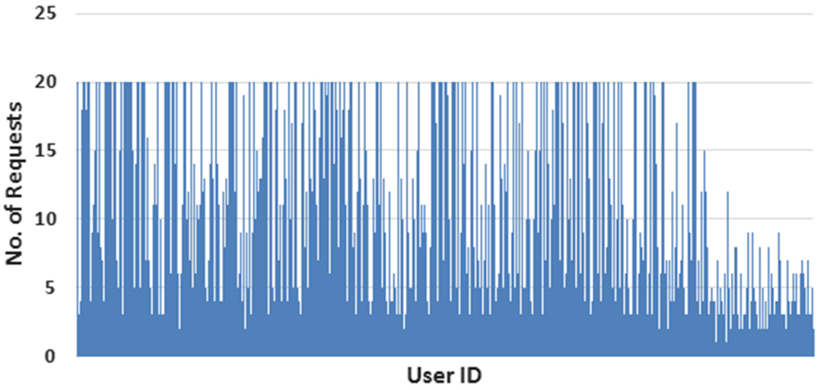


Fig. 6. The total number of requests made by each user

Figure 7 shows the average watching ratio for each user. Only 800 users have a departure with zero watching ratio and the reason behind such departure might be an explicit session closure or a system failure. Figure 8 shows the watching ratio discretized into ten levels versus total count. The first level (1–10%) has the higher number of users (26667) followed by level 10 (90–100%) which has 13508 users. The remainder of users (22921) has a watching ratio in between 20–90%.

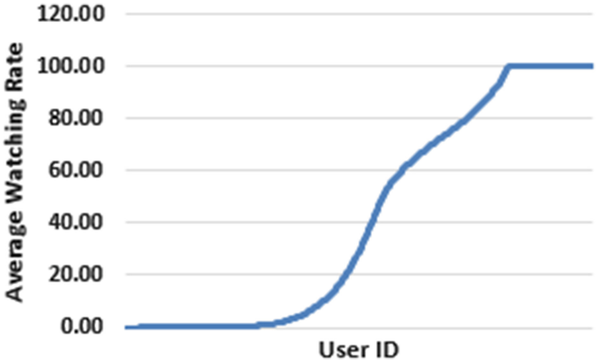


Fig. 7. Average watching ratio made by each user

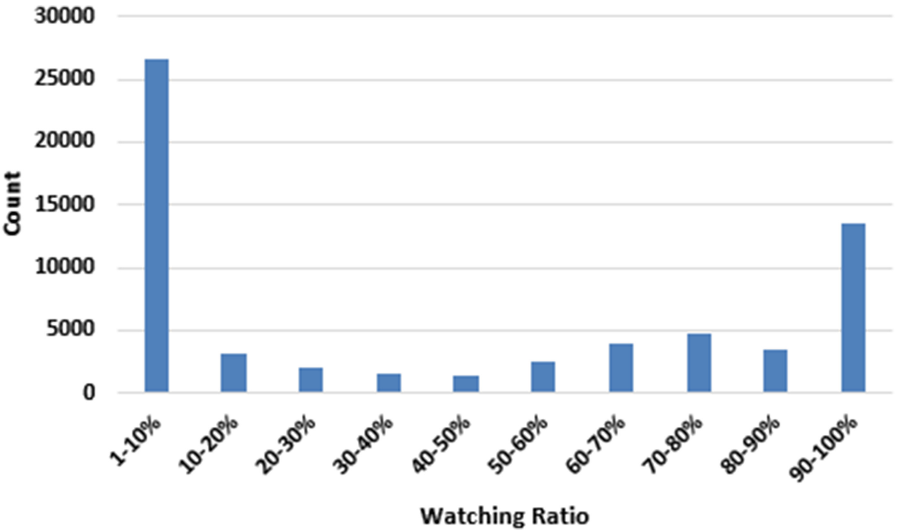


Fig. 8. Watching ratio levels vs. count

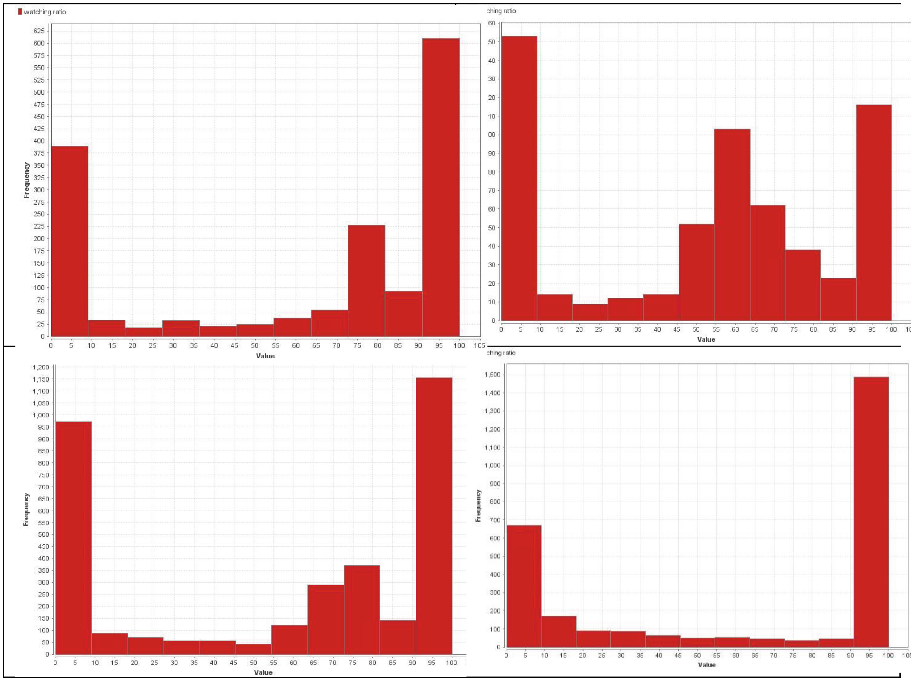


Fig. 9. Histogram of watching ratio per video type

Finally, we study the impact of video type on video watching time and characterize the video watching rate on a per-type basis. This section analyzes the relation between video watching rate and high ranked videos. As shown in Fig. 9, using the watching ratio histogram of four representative examples of top ranked videos, it is possible to appreciate a first distribution from the (0–10%) interval to (90–100%) and another around 100%. The reader can notice that a higher frequency of users achieve a watching ratio more than 50% of all 4 top-ranked videos.

5 Conclusion and Future Work

This paper presented statistical analysis of user behavior in a VoD service over a 6-months period. User behavior was analyzed from the real Lancaster Living Lab. The proposed analysis offers rich details into how users interact with such a converged service. The obtained results could be mined for different research insights in a myriad of research fields. For instance, P2P video delivery systems that takes into account different user's behaviors in a way that make it possible to deliver appropriate media content to members. Future work will address the implementation of the proposed recommender. Some participating peers called malicious might not cooperate as desired. They may be selfish and unwilling to cooperate as it should be. How to inhibit and detect those malicious peers in P2P networks is an open issue we are going to address. Besides, further research work will be on how to extract more interesting chunks in videos and only deliver these chunks with the dependence on user's behavior analysis.

References

1. Liu, Y., Guo, Y., Liang, C.: A survey on peer-to-peer video streaming systems. Peer-to-Peer Netw. Appl. **1**(1), 18–28 (2008)
2. Youtube Web Site: https://www.youtube.com/yt/press/statistics. Accessed 2 Jan 2015
3. Ramzan, N., Park, H., Izquierdo, E.: Video streaming over P2P networks: challenges and opportunities. Image Commun. **27**, 401–411 (2012)
4. Zheng, Y., Peng, J., Yu, Q., Huang, D., Chen, Y., Chen, C.: A measurement study on user behavior of P2P VoD system. In: The Proceedings of the 2nd International Asia Conference on Informatics in Control, Automation and Robotics, CAR 2010, Piscataway, NJ, USA, vol. 3, pp. 373–376, IEEE Press (2010)
5. Liao, X., Jin, H., Yu, L.: A novel data replication mechanism in P2P VoD system. Future Gener. Comput. Syst. **28**, 930–939 (2012)
6. Ma, K.J., Bartoš, R., Bhatia, S.: Review: a survey of schemes for internet-based video delivery. J. Netw. Comput. Appl. **34**, 1572–1586 (2011)
7. Hei, X., Liang, C., Liang, J., Liu, Y., Ross, K.: A measurement study of a large-scale P2P IPTV system. IEEE Trans. Multimedia **9**, 1672–1687 (2007)
8. Almeida, J.M., Krueger, J., Eager, D.L., Vernon, M.K.: Analysis of Educational Media Server Workloads. NOSSDAV, Port Jefferson (2001)
9. Chesire, M., Wolman, A., Voelker, G., Lavy, H.: Measurement and analysis of a streaming-media workload. In: USENIX Symposium on Internet Technologies and Systems (2001)
10. Wang, Y., Claypool, M., Zuo, M.: An empirical study of realvideo performance across the internet. In: ACM SIGCOMM Internet Measurement Workshop, San Francisco, USA, pp. 295–309 (2001)

11. Loguinov, D., Radha, H.: Measurement study of low-bit rate internet video streaming. In: ACM SIGCOMM Internet Measurement Workshop (IMV) (2001)
12. Costa, C., Cunha, I., Borges, A., Ramos, C., Rocha, M., Almeida, J., Ribeiro-Neto, B.: Analyzing client interactive behavior in streaming media servers. In: Proceedings of 13th ACM International World Wide Web Conference (WWW), New York City, NY, May 2004
13. Cherkasova, L., Gupta, M.: Analysis of enterprise media server workload: access patterns, locality, content evolution and rates of change. IEEE/ACM Trans. Netw. **12**, 781–794 (2004)
14. Acharya, S., Smith, B., Parnes, P.: Characterizing user access to videos on the World Wide Web. In: Proceedings of MMCN, January 2000
15. Arias, J.R., Suarez, F.J., Garcia, D.F., Panieda, X.G., Garcia, V.G.: Evaluation of video server capacity with regard to quality of the service in interactive news-on-demand systems. In: Protocols and Systems for Interactive Distributed Multimedia (PROMSIDMS2002), Coimbra, Portugal. LNCS, vol. 2515 (2002)
16. Veloso, E., Almeida, V., Meira, W., Bestavros, A., Jin, S.: A hierarchical characterization of a live streaming media workload. In: ACM Internet Measurement Workshop (IMV), November 2002
17. Sripanidkulchai, K., Maggs, B., Zhang, H.: An analysis of live streaming workloads on the internet. In: Proceedings of ACM Internet Measurement Conference 2004, Sicily, Italy, October 2004
18. Li, V., Liao, W., Qiu, X., Wong, E.: Performance model of interactive video-on-demand systems. IEEE J. Sel. Areas Commun. **14**, 1099–1109 (1996)
19. Garcá, R., Pañeda, X.G., Garcá, V.G., Melendi, D., Vilas, M.: Statistical characterization of a real video on demand service: user behaviour and streaming-media workload analysis. Simul. Model. Pract. Theory **15**(6), 672–689 (2007)
20. Elkhatib, Y., Mu, M., Race, N.: Dataset on usage of a live & VoD P2P IPTV service. In: Proceedings of the IEEE International Conference on Peer-to-Peer Computing (2014)
21. Elhoseny, H., Elhoseny, M., Abdelrazek, S., Bakry, H., Riad, A.: Utilizing Service Oriented Architecture (SOA) in smart cities. Int. J. Adv. Comput. Technol. (IJACT) **8**(3), 77–84 (2016)
22. Elhoseny, M., Yuan, X., Yu, Z., Mao, C., El-Minir, H., Riad, A.: Balancing energy consumption in heterogeneous wireless sensor networks using genetic algorithm. IEEE Commun. Lett. **19**(2), 2194–2197 (2015). http://dx.doi.org/10.1109/LCOMM.2014.2381226
23. Yuan, X., Elhoseny, M., Minir, H., Riad, A.: A genetic algorithm-based, dynamic clustering method towards improved WSN longevity. J. Netw. Syst. Manag. **25**(1), 21–46 (2017). http://dx.doi.org/10.1007/s10922-016-9379-7

Intelligent Decision Framework to Explore and Control Infection of Hepatitis C Virus

Mohamed M. Reda Ali[1](✉), Yehia Helmy[2], Ayman E. Khedr[3], and A. Abdo[3]

[1] Central Lab of Agriculture Expert Systems (CLAES),
ARC, Giza, Egypt
mreda@claes.sci.eg, m_reda25@yahoo.com
[2] Faculty of Commerce and Business Administration,
Helwan University, Cairo, Egypt
yehiahelmy@yahoo.com
[3] Faculty of Computers and Information, Helwan University, Cairo, Egypt
Ayman_khedr@helwan.edu.eg, amanyabdo_80@yahoo.com

Abstract. This research presents Intelligent Decision Framework (IDF) to explore and manage cases of hepatitis c virus based on data mining approach and Fuzzy logic system. The proposed framework is produced from integration between data mining decision tree, rule based classification and fuzzy logic system. On the other hand, this study improves the predication results of Fibrosis stage by using Trapezoidal Fuzzy Number (TFN) distribution as fuzzy logical system to arrive 98.1% compared to predication results that were 92.5% by data mining decision tree model for same patients sample. Fuzzy logic system predicts disease scale of Hepatitis C Virus (HCV) for patients sample through different stages of liver disease caused by virus c. The proposed framework supports physicians and Ministry of Health (MOH) strategies for treatment to limit and control HCV infections and prevalence rate in Egypt and other countries. The extracted knowledge and information from proposed framework helps decision makers to take appropriate and better decision at appropriate time to against hepatitis c viral in world. The architecture of intelligent decision framework is designed to support physicians to investigate and present treatment for HCV cases. Also, to develop intelligent machine, health care system or robots as a physician for HCV patients in high prevalence rate countries.

Keywords: Intelligent Decision Framework (IDF) · Hepatitis C virus (HCV)
Data mining (DM) · Decision tree · Fuzzy logic system (FLS) · Liver fibrosis
Intelligent decision support system (IDSS) · Biochemical markers

1 Introduction

130–170 million people are infected with HCV around the world with approximate percentage is 3% of world population. 3–4 million people are infected by HCV every year. The percentage of people infected with HCV in Egypt is 14.7%, representing

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 264–274, 2018.
https://doi.org/10.1007/978-3-319-74690-6_26

about 11 million persons. 90% of Egyptian patients are infected by HCV genotype 4. Where, HCV has 6 genotypes [1, 3]. Intelligent Decision Support Systems (IDSS) for health care is used to collect the required data and extract information and knowledge to support physicians to perform high accuracy diagnoses for patient cases and determine appropriate therapy. Intelligent Health Care Systems (IHCS) use different intelligent techniques such as data mining, fuzzy logic system, neural networks, expert system, case base reasoning…etc. [3, 4]. IDSS able to achieve business main goals, requirements and objectives by using the extracted information and knowledge through complex decision making processes. This study uses fuzzy logical system technique to enhance prediction process for IHCS which named biochemical markers of fibrosis for chronic liver disease based on data mining decision tree technique [3]. The outline of this paper is as follows. In section two, the paper begins with a brief background to illustrate Fuzzy Logic system (FLS) and facts about HCV. In section three, presents literature review and related work to explore biochemical markers and liver fibrosis technique, treatments of HCV and intelligent systems for health care. The prevalence of HCV infections in Egypt represents in section four. Section five discusses research methodology and proposed framework to explore and control HCV infections in section six. Case study in section seven and section eight presents discussion and evaluation about results of case study. Finally, conclusions and future work will be summarizing in section nine.

2 Background

An intelligent system achieves many complex tasks that carried out by humans and need amount of intelligence. Intelligent systems perform many processes such as screening, shifting, filtering for data to increase data overflow, supporting decision making processes to increase production and system effectiveness. An intelligent system is a smart module to increase organization production and useful for users [4, 5]. The following sub-sections explore the features of Fuzzy Logic system (FLS) in first section, the facts of HCV in second section and ways to transmit HCV in third section.

2.1 Fuzzy Logic System (FLS)

Fuzzy Logic (FL) is used to represent non-statistical imprecision and vagueness in information and data. FL applied in many areas such as engineering, medicine, decision analysis, and computer science [6, 7]. Fuzzy logic model composed of: fuzzifier/input, Fuzzy inference engine, fuzzy rule base and defuzzifier/output, that will be used by study to compose component of a proposed IDF [7]. A fuzzy set defined as a set that allows its member to have different degree of Membership Function (MF) in range between [0,1]. That represented as following: Let X is the universe of discourse and its element is denoted by x, then a fuzzy set A in X, which is denoted as A $\subseteq$ X, is defined as a set of ordered pairs. A = $\{(x, \mu A(x))| \ x \in X\}$; where $\mu A(x)$ is called MF of x in A, $\mu A(x)$: $X \rightarrow [0, 1]$, $0 < \mu A(x) < 1$ [6].

2.2 Facts About HCV

Number of infected people around the world is 130–170 million persons. Number of infected people by chronic HCV people in different countries as following: 17000–21000 persons in Denmark [2]. 214,000 case in United Kingdom (UK) [8], 1.5–2 million people in Japan [9], 280,000 case in Canada [9], 3.2 million persons in USA [10] and 11 million persons in Egypt [1, 3]. In USA 80% of infected persons by HCV genotype 1 and developed to chronic Hepatitis C then to liver scarring. About 15000 people die every year and at end stage of disorder may be caused liver cancer with percentage 1–5%, need a liver transplant operation or other patients will die [10]. 180000 enrolled Veterans in USA infected by HCV, 15% of them treated in last two years by budget 700 million USD. Treatment drugs for patients determined according to stage of HCV infection [11].

2.3 Ways to Transmit HCV

HCV is not transmitted through holding hands, sharing a glass or kissing. There are many ways to transmit HCV disorder such as: Infection blood products, Insufficient sterilized tools in medical surgical, dental operations and tattooing, sharing syringes and needles, sharing sniffing, unprotected sex with an infected menstruation, outbreaks of herpes on the genitals or anal sex, and children born with HCV from mothers. Symptoms of HCV different form patient to others but from some patients experience disease affected the liver: pain in the liver on bottom of the rib cage at the right side of the human body, pain or swelling in the abdomen, dark urine, fatigue, fever, itching, lack of appetite, nausea, arthritis and vomiting, pale stools or jaundice [10, 17].

3 Literature Review and Related Work

This research based on Hepatitis C virus (HCV) data that are collected from the following resources: HCV data and laboratory examinations [3]. HCV data in Egypt demographic and health survey 2008 and HCV treatment methods in Egyptian control strategy for viral hepatitis 2008–2012 [1, 15, 16].

3.1 Biochemical Markers and Liver Fibrosis

Biochemical markers and liver fibrosis tests in laboratory for HCV patients help doctors to diagnosis stages of HCV and determine treatment course according to infection degree or progress of liver fibrosis. Data mining process is used to extract information and patterns from data to build predictive models [3]. Sabry et al. (2013) constructed decision tree for liver biopsy by biochemical markers to assessed and identified risk of cirrhosis for HCV patients. Their study used training data to produce decision tree model by using data mining software which called WEKA to predict HCV infection degree or Fibrosis stage for patients. The predication results of Fibrosis stage by decision tree model were 92.5% compared to-results of biopsy samples. HCV diagnostic test depend on the result of liver biopsy which got a small sample of liver tissue the examined by pathologist under microscope to determine the degree of liver fibrosis or by fibrosis and activity tests. Also; Metavir scoring system (MSS) is used to evaluate and determine degree, activity, inflammation amount, fibrosis stage, or scarring of HCV liver according to ActiTes and FibroTest as a Biochemical Markers according to Metavir scoring system for liver biopsy to determine liver activity score and fibrosis score. Where, Liver activity or the degree of inflammation stage (A_0 for no activity, A_1 for minimal activity, A_2 for moderate activity, A_3 for severe activity) related with liver fibrosis score (F_0 for no fibrosis, F_1 for portal fibrosis without septa, F_2 for portal fibrosis with few septa, F_3 for numerous septa without cirrhosis, F_4 for cirrhosis). FibroTest measures the degree of fibrosis as shown in Table 1. Where, range of FibroTest unit from 0 for no fibrosis to 1 cirrhosis HCV [3].

Table 1. Conversion between FibroTest and fibrosis stage [3]

FibroTest	Metavir fibrosis stage estimate	ActiTest	Metavir activity grade estimate
0.75–1.00	F4	0.63–1.00	A3
0.73–0.74	F3–F4	0.61–0.62	A2–A3
0.59–0.72	F3	0.53–0.60	A2
0.49–0.58	F2	0.37–0.52	A1–A2
0.32–0.48	F1–F2	0.30–0.36	A1
0.28–0.31	F1	0.18–0.29	A0–A1
0.22–0.27	F0–F1	0.0–0.17	A0
0.00–0.21	F0		

3.2 Treatments of HCV

Ministry of Health (MOH) in Egypt established a committee for Control of Viral Hepatitis with budget 80 million USD/year. At 2012 HCV committee treated 20% of patients with combination of pegylated interferon (peg-INF) and ribavirin (RBV). Until now the cost of HCV treatment is high cost. Now HCV committee has about 40 centers to present two treatment regimens for 50,000 patients as follow: pegylated interferon (peg-INF), ribavirin, sofosbuvir for 3 months and sofosbuvir + ribavirin for 6 months [1].

3.3 Intelligent Systems for Health Care

Salih, A. and Abraham, A. (2015) developed IDSS to manage hospital emergency situations and reduce false alarm rate, number of vital attributes for patients and increase the sensitivity rate by evaluating processes for nine classifier algorithms. This is a health care IDSS used to monitor chronic diseases based on data mining classification technique by following algorithms (IBk, Attribute Selected Classifier,…etc.) [12]. Also, Abraham (2003) designed framework for hybrid intelligent systems based on hybrid soft computing architecture that integrated between Neural Networks (NN), Fuzzy Inference Systems (FIS), and Evolutionary Computation (EC) [13]. A Fuzzy Number (FN) is defined in the real set R, and in addition, is normal and convex and has a continuous Member Function (MF). The formal definition for FN is a fuzzy set A that defined on the set of real numbers, where $A \subseteq R$, MF represented as $\mu_A(x)$: $R \rightarrow [0,1]$ to satisfy conditions of FN function or shape. There are many type of FN such as

$$\mathrm{Trapezoid}(x:a,b,c,d)=\begin{cases}0 & x<a\\ \dfrac{(x-a)}{(b-a)} & a\le x\le b\\ 1 & b\le x\le c\\ \dfrac{(d-x)}{(d-c)} & c\le x\le d\\ 0 & x>d\end{cases}$$

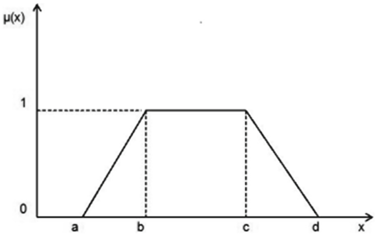


Fig. 1. MF of trapezoidal fuzzy number

Triangular, Trapezoidal, Gaussian, Exponential, Bell-shaped FN,… etc. Trapezoidal Fuzzy Number (TFN) named according to trapezoidal shape to represent MF as shown in Fig. 1 and define trapezoidal fuzzy number A as $\mu_A(x)$ or Ax = (a, b, c, d), where $a < b \leq c < d$ [6].

4 Problem Definition and Induction

HCV is a blood disorder caused inflammation for human liver and led to damages liver cells. HCV transfer from the infected blood through operations or personal tools [2, 10]. In the recent time, the current situation for the prevalence of HCV infections in Egypt as the following: 10% from Egyptian people infected by HCV genotype 4 and other 5% had HCV antibodies in blood. Where, Infection new cases are170000 persons per year, and annual death cases are 40000 persons per year. HCV prevalence rates in rural areas are 12% and urban areas are 7% [1, 3]. This research summarizes many problems in the following points according to strategies of Egyptian ministry of Health and current situation of infection, prevalence hepatitis C inflammation in Egypt [1, 15, 16]:

- There is no trace process for spread of HCV.
- There is not enough control and treatment budget to limit HCV infection.
- There is not sufficient awareness for the prevention of viral hepatitis C.
- Treatment services are offered to about 3% of HCV patients.

5 Research Methodology

Our research methodology of study based on the intelligent decision support system techniques as health care systems. Our research will use fuzzy logical system technique to enhance prediction process for IHCS which presented in biochemical markers of fibrosis for chronic liver disease study based on data mining decision tree technique. Also, it will present the required recommendations and heuristic judgments to monitor and control Infection of HCV in Egypt. The steps of research methodology:

- Data preparation step to collect patients data that related to patient blood sample, biochemical markers and liver fibrosis tests in laboratory for HCV patients as records lab fibrosis score result $\{F_0, F_1, F_2, F_3, F_4\}$.

- Use data mining classification approach such as decision tree (ID3, C4.5 algorithms) [3]. then using rule base classification to extract set of classification rules (accuracy percentage) to predict class {F_0, F_1, F_2, F_3, F_4} of fibrosis stage instance based on biochemical markers and liver fibrosis tests in laboratory for HCV patients as main goal to find a model for fibrosis score and infection stage. Where, R = {r_1 v r_2 v r_3 v …..r_n}.
- Transform the interval and stage of fibrosis functions in Table 1 for patient sample in research case study to represent it by using TFN function set. Then new scale of TFN function for HCV patient solve the problem of many mathematical equations that used to calculate fibrosis score in laboratory to determine HCV infection stage {F_0, F_1, F_2, F_3, F_4} or intermediate fibrosis score such as {F_0-F_1, F_1-F_2, F_3-F_4} based on different investigation parameters. The name of investigation parameters is serum biochemical markers such as {alpha2-macroglobulin, haptoglobin, apolipoprotein A1, total bilirubin, gamma glutamyl transpeptidase (GGT), and alanine aminotransferase (ALT)…etc.} [3].
- Design intelligent decision framework to monitor, control infections of HCV, support treatment physicians and MOH strategies with current situation and results to determine treatment methodology and limit prevalence rate of HCV.

6 Proposed Framework

The proposed model for intelligent decision framework as IHCS to monitor, limit and control Infection of HCV in Egypt consist of the following components:

6.1 Problem Definition and Objectives

The research questions and objectives are summarized as the following:

- How to control and limit HVC?
- What are the required budget for treatment and control HCV infection in Egypt?
- What is the architecture of the proposed framework to control HCV Infection?
- What are the impacts of the proposed framework?
- How to enhance prediction process for biochemical markers of fibrosis for chronic liver disease study which applied data mining decision tree technique?

6.2 Database Component

This component is used to save healthcare data for Egyptian as electronic health recodes to support the medical services, operations and healthcare strategies for treating patient, limiting and controlling epidemic diseases such as HCV, HBV, Cancer,…etc.

6.3 Deep Analysis Component

This component is used to analysis patients data from electronic health recodes with current HCV Symptoms and laboratory investigations to support physicians in diagnosis and treatment operations to determine the appropriate therapy based on previous sub-steps in research methodology.

6.4 Protection and Intelligent Treatment Decision

This component is used to collect data about diseases to support healthcare strategies, physicians and patients with awareness information for public, proactive and predictive methodology for diseases diagnosis and treatment operations. Figure 2 illustrates the architecture of the proposed framework to explore and control Infection of HCV according the previous sub-sections as an Intelligent Health Care System (IHCS).

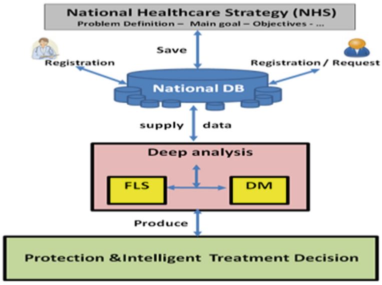

National Healthcare Strategy (NHS)
Problem Definition – Main goal – Objectives - ...
Save
Registration
Registration / Request
National DB
supply
data
Deep analysis
FLS
DM
Produce
Protection &Intelligent Treatment Decision

Fig. 2. Architecture of IHCS to explore and control HCV infections.

7 Case Study

This research based on HCV data that are collected from the following resources:

- HCV data and laboratory examinations for HCV patients sample that included 200 patients as following: 38 persons negative HCV infection, 4 patients had positive HCV genotype1 and 158 patients had positive HCV genotype 4 (123 males and 35 females) the range of the patients age from 23 to 69 years [3].
- HCV data in Egypt demographic and health survey (EDHS) which conducted in 2008. EDHS covered age range for group from 15 to 59 years for 6290 women, 5718 men to measure knowledge and awareness about HCV and other diseases. EDHS concluded that 79.9% of women and 85.4% of men know about hepatitis C [15].
- The study points to treatment methods for HCV cases in Egyptian national control strategy for viral hepatitis 2008-2012 and other treatment course for HCV in Egypt and USA [1, 11, 15, 16].

Table 2 shows rule base classification that extracted set of classification rules to predict class of fibrosis stage instances based on data mining decision tree variables or features for HCV cases, where the predictive attributes are Haptoglobin (H), Alpha2-macroglobulin (A), alanine aminotransferase (T), total bilirubin (TB), Gamma glutamyl transpeptidase (G), Albumin (B), Triglyceride (TG), Platelet (P), White blood cell count (W), Creatinine (C) and Age (E) [3, 14]. Our study defines five member functions of fibrosis stages (F_0, F_1, F_2, F_3, F_4) for HCV in Egypt and represent it by

Table 2. Classification, clustering and association rules

Classification rules	Clustering and association rules
r_1:(H <=0.984, M <=1.93) → F_0 r_2:(H <=0.984, M > 1.93,T <=24) → F_1 r_3:(H <=0.24, M > 1.93,T > 24) → F_3 r_4:(H > 0.24, M > 1.93,T > 24,TG <=233) → F_3 r_5:(H <=0.24, M > 1.93,T > 24, TG > 233) → F_2 r_6:(H <=0.984, M > 3.05, G <=13) → F1 r_7:(H <=0.984, M > 3.05, G > 13, TB <=8.55) → F_3 r_8:(H <=0.984, M > 3.05, G > 13, TB > 8.55) → F_4 r_9:(H > 2.36,M <=2.23) → F_2 r_{10}:(H <=2.36, M <=2.23) → F_0 r_{11}:(H > 0.984, M > 2.23, T > 65) → F_4 r_{12}:(H > 0.984, M <=3.39, E <=46) → F_1 r_{13}:(H > 0.984, M > 3.39, E <=46) → F_3 r_{14}:(H > 0.984, M > 2.3,T <=65, E > 46, B <=2.3) → F_4 r_{15}:(H > 0.984, M > 3.61,T <=65, E > 46, B > 2.3,P <=67.6) → F_3 r_{16}:(H > 0.984, M <=3.61,T <=65, E > 46, B > 2.3,P <=67.6) → F_2 r_{17}:(H <=1.68, M <=2.3,T < 65,E > 46,B > 2.3, P > 67.7,W <=7.7) → F_2 r_{18}:(H <=1.68, M <=2.3,T < 65,E > 46,B > 2.3, P > 67.7,W > 7.7) → F_0 r_{19}:(H > 1.68, M <=2.3,T < 65,E > 46,B > 2.3, P > 191) → F_0 r_{20}:(H > 1.68, M <=2.3,T < 65,E > 46,B > 2.3, P <=191,C <=2) → F_1 r_{21}:(H > 1.68, M <=2.3,T < 65,E > 46,B > 2.3, P <=191, C > 2) → F_2	Cluster 1 for F_0 patients F_0 = (r_1 v r_{10} v r_{18} v r_{19}) Cluster 2 for F_1 patients F_1 = (r_2 v r_6 v r_{12} v r_{20}) Cluster 3 for F_2 patients F_2 = (r_5 v r_9 v r_{16} v r_{17} v r_{21}) Cluster 4 for F_3 patients F_3 = (r_3 v r_4 v r_7 v r_{13} v r_{15}) Cluster 5 for F_4 patients F_4 = (r_8 v r_{11} v $r_{14)}$ Where F_0 has any combination from the following association rules F_0 = (r_1 v r_{10} v r_{18} v r_{19}) F_0 = (r_1) v (r_{10} v r_{18} v r_{19}) F_0 = (r_1 v r_{10}) v (r_{18} v r_{19}) F_0 = (r_1 v r_{10} v r_{18}) v (r_{19})…etc.

Table 3. Equations of HCV member functions of fibrosis stages by using TFN

Equations of TFN for fibrosis stages		
$\mu_{F0}(x)$ or $Ax = (0, 0, .21, .28)$ $\mu_{F0}(x) = \begin{cases} 1 & 0 \le x \le .21 \\ \frac{(d-x)}{(d-c)} & 0.21 \le x \le .28 \\ 0 & x \le .28 \end{cases}$		$\mu_{F4}(x)$ or $Ax = (.72, .75, 1, 1)$ $\mu_{F4}(x) = \begin{cases} 0 & x \le .72 \\ \frac{(x-a)}{(b-a)} & .72 < x < .75 \\ 1 & x \ge .75 \end{cases}$
$\mu_{F1}(x) = (.21, .28, .31, .49)$ $\mu_{F1}(x) = \begin{cases} 0 & x \le .21 \\ \frac{(x-a)}{(b-a)} & .21 < x < .28 \\ 1 & .28 \le x \le .31 \\ \frac{(d-x)}{(d-c)} & 0.31 < x < .49 \\ 0 & x \ge .49 \end{cases}$	$\mu_{F2}(x) = (.31, .49, .58, .59)$ $\mu_{F2}(x) = \begin{cases} 0 & x \le .31 \\ \frac{(x-a)}{(b-a)} & .31 < x < .49 \\ 1 & .49 \le x \le .58 \\ \frac{(d-x)}{(d-c)} & 0.58 < x < .59 \\ 0 & x \ge .59 \end{cases}$	$\mu_{F3}(x) = (.58, .59, .72, .75)$ $\mu_{F3}(x) = \begin{cases} 0 & x \le .58 \\ \frac{(x-a)}{(b-a)} & .58 < x < .59 \\ 1 & .59 \le x \le .72 \\ \frac{(d-x)}{(d-c)} & 0.72 < x < .75 \\ 0 & x \ge .75 \end{cases}$

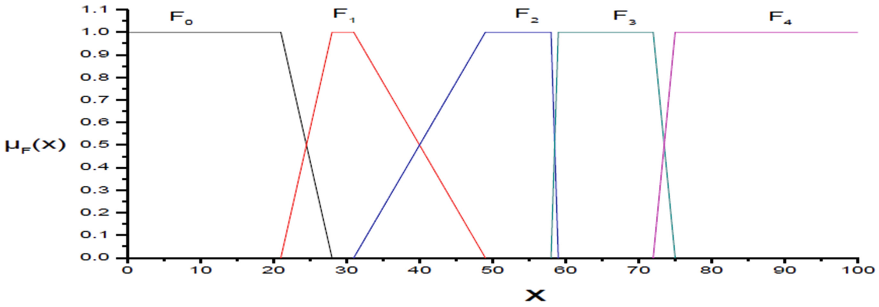


Fig. 3. TFN represents stages and values of FibroTest for HCV

TFN shape and define as follows: trapezoidal fuzzy number A as $\mu_{F(x)}$ or Ax = (a, b, c, d) with a $\leq$ b $\leq$ c $\leq$ d. Figure 3 illustrates the graphical representation for member functions of fibrosis stages by using TFN. Also, TFN equations for fibrosis stages of HCV in Table 3.

8 Discussion and Evaluation

Our study distributes eight stages of FibroTest for patient sample as shown in Fig. 3 and Table 4. Also it applies the average distribution for intermediate stages F_0-F_1, F_1-F_2 to divide number of patients in current interval between previous and next intervals as the following: For the case of the number of patients sample which is even (n) for intermediate interval, the average distribution which added to both intervals is n/2. Otherwise, add (n + 1) / 2 as an average distribution to interval which has low number of patients and (n - 1) / 2 to another interval. For the case of intermediate interval F_3-F_4, the study will add total population number to the previous interval F3. Where, the patients in the previous interval not exactly arrive to last infection case. Figure 4 illustrates the study distribution for FibroTest results by using TFN compared with Fibro stages in liver biopsy for 158 patients. The results of Fibro Test (FT), fibro stages in liver biopsy and TFN distribution for Fibro results for patients sample (n = 158) as shown in Table 4. Also, the predication results of Fibrosis stage by using TFN distribution is 98.1% compared to results of biopsy samples (n = 158).

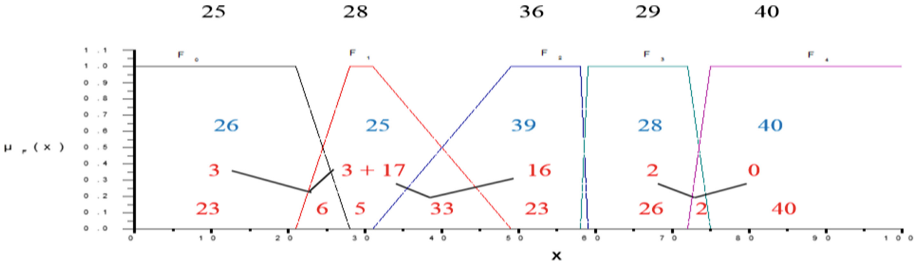


Fig. 4. TFN representation and distribution for Fibro stages based on FibroTest results.

Table 4. TFN distribution for Fibro-Test, Fibrosis stages in liver biopsy and FibroTest results.

Fibro Stage	FibroTest (FT) () Patient No. (n = 158)	TFN distribution for FT (n158)	Compared %	Fibro stages in liver biopsy
F0	23	26	+1 (+ 0.6%)	25
F0–F1	6	–		–
F1	5	25	−3 (−1.9%)	28
F1–F2	33	–		–
F2	23	39	+3 (+1.9%)	36
F3	26	28	−1 (− 0.6%)	29
F3–F4	2	–		–
F4	40	40	0 (0%)	40

According to study results, MOH in Egypt needs proactive policies and protocols to control and limit HCV Infection in Egypt to decrease, prevent new infection case by HCV and limit the prevalence of it. Also, increase awareness and treatment budgets. An electronic health care recodes can be developed to control epidemic diseases in uncertainty environment such liver hepatitis, hearts, cancers…etc.

9 Conclusions and Future Work

The challenge does not only in cost of treatment, but also in diagnosis which remains inadequate given the high illiteracy rates and low HCV awareness levels. Our study improves the predication results of Fibrosis stage by using TFN distribution with percentage 98.1% compared to predication results that were 92.5% by data mining decision tree model for same biopsy samples. Also, this study presents intelligent decision framework based on Fuzzy logic system to monitor and control infection of hepatitis c virus. Fuzzy logic system predicts HCV disease scale and percentage for patients sample according to different stages of liver disease. IDF supports physicians and MOH strategies to limit and control HCV infections and prevalence rate in Egypt. The study concludes the following recommendations to support MOH and proactive National Healthcare Strategy (NHS) to control and limit the prevalence of HCV infections in Egypt at current and future times: Increase public awareness by using TV, internet, universities…etc. and treatment budget. Design Integrated MIS as an electronic health recodes and health care data warehouse will help in controlling epidemic diseases and health care for Egyptian public. Implement the previous points to prevent new HCV infection cases in recent and future times. Also, it will lead to high control on public environment and medical tools which causes HCV infection. In future, study will work to generalize this framework with others human diseases such as heart and cancer diseases…etc.

References

1. Gaber, M.: HCV Treatment in Egypt, Economic and Social Justice Unit (2014). https://www.eipr.org/sites/default/files/pressreleases/pdf/hcv_treatment_in_egypt.pdf. Accessed 25 Nov 2017
2. Springborg, M.: Healthcare – A Growing Industry. https://docfinder.is.bnpparibas-ip.com/api/files/7B6261B5-0809-4693-8D58-B8016A4F276F. Accessed 25 Nov 2017
3. Sultan,T., Khedr, A., Sabry, S.: Biochemical Markers of Fibrosis for Chronic Liver Disease: Data Mining-Based Approach, Master thesis, FCI, Helwan University, Egypt (2013)
4. Kaklauskas, A.: Biometric And Intelligent Decision Making Support, Intelligent Systems Reference Library, vol. 81. Springer, Switzerland (2015)
5. Jantan, H., Hamdan, A., Othman, Z.: Intelligent techniques for decision support system in human resource management, chap. 16, InTechOpen (2010). https://cdn.intechopen.com/pdfs-wm/10951.pdf. Accessed 25 Nov 2017
6. Elomda, B., Hefny, H., Hasaan, H.: A Soft Computing Approach for Multi Criteria Decision Making System, Master thesis, ISSR, Cairo University, Egypt (2012)
7. Osofisan, P.: Fuzzy logic control of the syrup mixing process in beverage production. Leonardo J. Sci. **6**(11), 93–109 (2007)
8. Hepatitis C in the UK: 2017 report. https://www.gov.uk/government/publications/hepatitis-c-in-the-uk. Accessed 25 Nov 2017
9. Hepatitis C around the world. http://hcvadvocate.org/publications/fact-sheets/hcsp-fact-series/hepatitis-c-around-the-world-facts/. Accessed 25 Nov 2017
10. Hepatitis central in USA. http://www.hepatitiscentral.com/. Accessed 25 Nov 2017
11. Veteran Treatment. http://www.natap.org/2015/HCV/hepatitis_c_treatment_summary.pdf. Accessed 25 Nov 2017
12. Salih, A., Abraham, A.: Intelligent decision support for real time health care monitoring system. Advances in Intelligent Systems and Computing, Vol. 334, pp. 183–192. Springer, Cham (2015)
13. Abraham, A.: Intelligent systems: architectures and perspectives. Recent Advances in Intelligent Paradigms and Applications, pp. 1–36, Springer, Heidelberg (2003)
14. Ranka, S.: Data Mining. https://www.cise.ufl.edu/class/cis4930fa15idm/index.html. Accessed 25 Nov 2017
15. El-Zanaty, F., Way, A.: Egypt Demographic and Health Survey 2008, Ministry of Health. https://dhsprogram.com/pubs/pdf/FR220/FR220.pdf. Accessed 25 Nov 2017
16. Egyptian National Control Strategy for Viral Hepatitis 2008–2012, Ministry of Health (2009). http://www.thebera.eg.net/images/pdf/article1.pdf. Accessed 25 Nov 2017
17. Transmission of Hepatitis C. http://www.epidemic.org/thefacts/hepatitisc/transmission/. Accessed 25 Nov 2017

Supervised Rainfall Learning Model Using Machine Learning Algorithms

Amit Kumar Sharma(✉), Sandeep Chaurasia, and Devesh Kumar Srivastava

Manipal University, Jaipur, India
amitchandnia@gmail.com, chaurasia.sandeep@gmail.com, devesh988@yahoo.com

Abstract. Unpredictable and uncertain volume of the rainfall is the serious nature disaster. In current, available rainfall forecasting model predict rainfall volume hourly, weekly or monthly. This work proposed a supervised learning model which is based on machine leaning algorithms of data mining. This approach classify the low, mid and high volume of rainfall. Proposed approach is practically implemented on different uncertain heavy rainfall regions and compare the accuracy and measured the accuracy by ROC area of classifiers such as Random Forest, SMO, Naive Bayes and Multilayer Perceptron (MLP).

Keywords: Supervised learning · Normalization · Mean Thresholds · Rainfall dataset · Feature selection · Machine learning Naive Bayes · Random forest · MLP · SMO

1 Introduction

Uncertain heavy rainfall is the serious natural disaster such as flooding, landslide or debris flows. Uttarakhand state of India has faced worst natural disaster in June 2013. There was received approx 400% more rainfall compare to normal monsoon rainfall. Due to such type of heavy rainfall, bridges and roads was destructed and 100,000 pilgrims and tourists trapped which was on "Char Dham Yaatra" [9]. Indian Meteorological Department (IMD) could not predict such amount of heavy rains, causing thousands of people loss their life and property. Indian Meteorological Department is soon going to take a call on shifting from a statistical model of rainfall predicting to the coupled dynamical model, deemed to be more accurate and run on a supercomputer [10]. IMD issues its forecast for the June-September monsoon season in April, which is based on the ensemble statistical forecasting system that uses five predictors. On other hand, dynamic forecasting system is based on current physical observations of the atmosphere, cloud properties and oceans, which can then be used to get a forecast [10]. Artificial neural network is playing a key role in current research. Back propagation ANN has been successfully implemented to built a rainfall pattern recognition [1]. Feed forward neural network also was used for the same concept to build another rainfall forecast model. These all the rainfall forecasting models work with rainfall records in the last period of time as the input

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 275–283, 2018.
https://doi.org/10.1007/978-3-319-74690-6_27

data, when data gaps occur these models might not work properly. All forecasting model predict rainfall hourly, weekly or monthly. Scientist has been running experiments to develop a more accurate and more reliable model for weather forecast specially rainfall and cloudburst. This study intends to build a model with better accuracy for rainfall forecast. Proposed model is good for its better designed function which include normalization, average and thresholds in the given datasets. Thresholds are used to categorization of the rainfall dataset. Paper describes the supervised learning approach to classify the given rainfall data and it also define the accuracy of the model. Random forest, Naive Bayes, SMO and MLP machine learning algorithms are used to classify the data.

2 Related Work

During literature survey, a unsupervised method is adopted to provide accurate and effective typhoon hourly rainfall forecast [1]. Performance were tested by coefficient of and coefficient of efficiency [1]. Genetic programming were applied for Indian summer monsoon rainfall predication [2]. Drawback of this proposed method does not consider temporal changes in the relationships [2]. A dynamic model have been applied for pattern recognition and prediction through artificial neural network technique [3]. Machine learning technique, SVM is used to predict rainfall for hourly prediction [4]. Back propagation neural network were applied for daily rainfall, monthly rainfall and extract rainfall features [5]. Qualitative and quantitative methods and Mann Kendall test method was used to data analyzed [6]. Mann-Kendall trend test, markov method and baysian analysis were used to find pattern in the rainfall [7]. A procedure combined, Mann-Kendall trend test, spatial mapping techniques and wavelet transformed analysis were used to identify the patterns. Statistical analysis were used for rainfall data, the mean, variance, coefficient of variation, Pearson's correlation was determined for datasets [8].

3 Problem Formulation

Uncertain heavy rainfall, have serious disaster such as landslide & flood, these disasters affects on humans life and property. To reduce such type of disasters, we need to build better disaster warning system in which rainfall prediction system is the most important.

4 Proposed Model

Proposed model is based on data mining approach for classifying low, mid & high volume of prediction to make more effective to make prediction system. Data mining process data selection, pre-processing, classification and evaluation process are applied for the better results. The proposed model is has been successfully applied on different datasets of rainfall. The proposed working model is showing in Fig. 1:

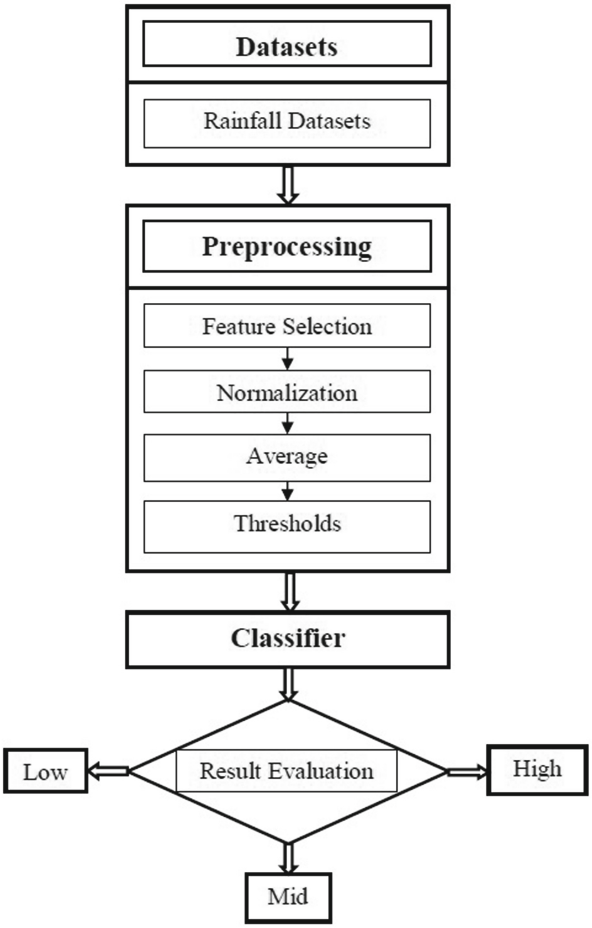


Fig. 1. Proposed working model for low, mid & high rainfall volume classification

4.1 Data Selection

For experiment purpose, we have used four regions rainfall volume of data which is measured in millimeter (mm) from Almora, Bhageshwar, Chamoli and Tehri Garhwal of Uttarakhand state of north India. These data sets have January to December rainfall volumes data of 102 years [12].

4.2 Preprocessing

Data origins from different sources and the data may be noisy, incomplete, inconsistent and missing. There is required to clean, transform and reduction in the data for better results. Steps are applied for data pre-processing:

Feature Selection: Feature selection is refers to most meaningful inputs for processing and analysis or extract useful information from existing large data. In this paper for experiment purpose, monthly rainfall volume in mm is being used.

Normalization: During data preprocessing step, data normalization or feature scaling method is used for the standardize the range of independent of variables. For the re−scaling between [0, 1] or [1, 1], we applied the general formula is given as:

$$x' = [x - min(x)]/[max(x) - min(x)]$$

where x is an original value, x' is the normalized (re−scale) value.

Calculate Average: The most widely used method of calculating an average is the 'mean'. The addition of the values divided by the total number of values. In this paper for analysis, we calculate the average of the January to December monthly rainfall volumes for every 102 years.

Calculate Thresholds: For data identifiers over a range of values, we calculate thresholds. Threshold needs when the values falls within selected range. In our method, we calculate two different thresholds for low, mid and high volume range of the rainfall datasets.

4.3 Classification Technique

Classification technique of data mining classifies the large volume of data into separate classes. For identifying good classifier, we are working with Random Forest, Naive Bayes, MLP & SMO classifier to classify low, mid & high classes of the particular rainfall of the different regions.

Naive Bayes Classifier: Naive bayes classification is a statistical analysis approach, which help in predicting the probability of class membership. Naive bayes technique is based on Bayes' theorem which provides posterior probability p(c|x) from P(c), P(x) & P(x|c):

$$P(c|x) = \frac{P(x|c)P(c)}{P(w)}$$

Random Forest Classifier: Random Forests technique is supervised learning method which applied to generate many classification trees. This method puts every input vector at bottom in each tree of the forest to classify a new object from an input feature vectors. Each tree in forest provides the "votes" to each tree and tree with highest "votes" are considered for classification. pseudo−code [11]:

1. Random choose “k” features from total “m” features. (k<m)
2. We calculate the element “x” from the “k” features by using the best split.
3. Split the element into son elements by using the best split.
4. Repeat 1 to 3 steps until “l” number of element has been reached.
5. Generate forest by repeating steps 1 to 4 for “n” number of times to create “n” number of trees.

Sequential Minimal Optimization (SMO): SMO is an iterative algorithm for solving the optimization problem that arises during the training of Support Vector Machine (SVM). SMO breaks this problem into series of possible sub problems [4].

Multilayer Perceptron (MLP): MLP is a class of feed forward ANN. An MLP consists of at least three layers of nodes. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. MLP use a supervised learning technique called back propagation for training purpose [3]. Three layer architecture of MLP Fig. 2:

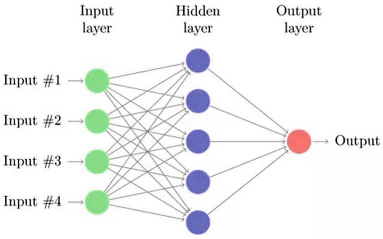


Fig. 2. Three layer architecture of MLP

4.4 Data Analysis

Evaluation of results from different classifiers which have been applied on different rainfall datasets. We have analyzed the our results to find out correctly classified instances (CCI) and incorrectly classified instances (ICCI), ROC area, accuracy and confusion matrix. Different parameters are as follow:

True Positive (TP): Correctly Low, Mid & High volume of rainfall detected rate that is actual Low, Mid & High volume of rainfall.

False Positive (FP): Incorrectly Low, Mid & High volume of rainfall detected rate that is not Low, Mid & High volume of rainfall.

Accuracy = (Correctly Classified Instances/Total Instances) * 100

ROC Area: ROC (Receiver Operating Characteristics) curves were developed to set the parameters (before learning) or thresholds (after learning) of a many learning algorithms for any 2 classes. The ROC curve is a plot of True Positive Rate against False Positive Rate.

Confusion Matrix: For the evaluation of classifiers, the confusion matrix provides the results of classifier's performance by displaying the correctly and incorrectly classified instances.

Confusion Matrix

$$\begin{array}{cccl} a & b & c & < - - classifiedas \\ 50 & 1 & 0 & |a = Mid \\ 0 & 49 & 0 & |b = Low \\ 2 & 0 & 0 & |c = High \end{array}$$

5 Experimental Evaluation

5.1 Setup for Experiment

Experimental rainfall data sets are related to heavy rainfall regions of India. We apply our strategy on rainfall volumes of four different regions which are Almora, Bhageshwar, Chamoli and Tehri Garhwal.

5.2 Experiment Results

Proposed successfully applied on four region's volume of rainfall datasets. We get the preprocessed results after calculate average and thresholds of the January to December data of the 102 years. For classifying the data, applied Naive Bayes, Random Forest, SVM and Multilayer Perceptron classifier. Results were observed as Tables 1, 2, 3 and 4:

Table 1. Results: Apply Naive Bayes (NB) algorithm on different regions of India

Regions	Correctly classified instances	Incorrectly classified instances	True positive rate (%)	False positive rate (%)	Accuracy (%)	ROC area
Almora	99	3	97.1	2.9	**97.1**	0.965
Bhageshwar	99	3	97.1	3.1	**97.1**	0.964
Chamoli	98	4	96.1	3.7	**96.1**	0.981
Tehri Garhwal	100	2	98.0	2.7	**98.0**	0.989

Table 2. Results: Apply random forest classifier on different regions of India

Regions	Correctly classified instances	Incorrectly classified instances	True positive rate (%)	False positive rate (%)	Accuracy (%)	ROC area
Almora	100	2	98.0	2.0	**98.0**	0.990
Bhageshwar	101	1	99.0	1.0	**99.0**	0.990
Chamoli	101	1	99.0	1.0	**99.0**	1.000
Tehri Garhwal	101	1	99.0	1.3	**99.0**	0.988

Table 3. Results: Apply Multilayer Percepron (MLP) classifier on different regions of India

Regions	Correctly classified instances	Incorrectly classified instances	True positive rate (%)	False positive rate (%)	Accuracy (%)	ROC area
Almora	100	2	98.0	2.0	**98.0**	0.980
Bhageshwar	100	2	98.0	2.0	**98.0**	0.964
Chamoli	99	3	97.1	2.9	**97.1**	0.996
Tehri Garhwal	99	3	97.1	4.0	**97.1**	0.988

Table 4. Results: Apply Sequential Minimal Optimization (SMO) classifier on different regions of India

Regions	Correctly classified instances	Incorrectly classified instances	True positive rate (%)	False positive rate (%)	Accuracy (%)	ROC area
Almora	95	7	93.1	6.9	**93.1**	0.931
Bhageshwar	96	6	94.1	6.1	**94.1**	0.940
Chamoli	98	4	96.1	3.9	**96.1**	0.966
Tehri Garhwal	93	9	91.2	12.1	**91.2**	0.895

5.3 Result Evaluation

Experiments result of Naive Bayes, Random Forest, MLP & SMO classifiers are compared and observed that Random Forest classifier provides the better accuracy for all the regions Fig. 3.

The reason of the better accuracy from random forest classifier is depends on generated number of decision trees. These trees amalgamate together and gives the more accurate prediction. Weka explorer bag with 100 iterations default which generate 100 number of trees for random forest classifier.

Accuracy is measured by the ROC area. ROC area of 1 represents a perfect test, an ROC area of 0.5 represents a worthless test. Our accuracy of ROC area is classified between 0.90−1 which represent excellent test.

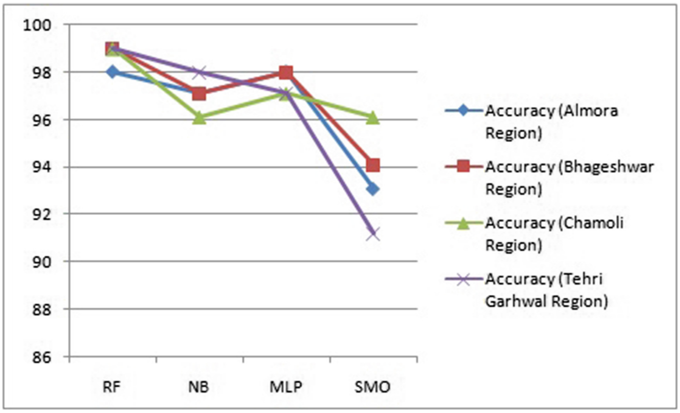


Fig. 3. Accuracy graph for all the regions with different classifiers

6 Conclusions and Future Work

Paper has presented a supervised rainfall learning model which has used machine learning algorithms for classifying the low, mid and high volume range of rainfall data. Presented approach have been successfully applied on different rainfall datasets of four Indian's regions. We have compared its accuracy and accuracy has measured by ROC area with the Naive Bayes, Random Forest, SMO and MLP classifiers. Proposed approach acquired to higher accuracy rate that is 99% for all rainfall datasets. This model provides better accuracy for supervised training data, future work will be based on unsupervised learning for finding better pattern recognition system with more features like temperature, moisture and geological study etc.

References

1. Lin, F.-R., Wu, N.-J., Tsay, T.-K.: Applications of cluster analysis and pattern recognition for typhoon hourly rainfall forecast. Eur. J. Sci. Res. Hindawi, Advances in Meteorology **2017**(5019646), 17 (2017). https://doi.org/10.1155/2017/5019646
2. Kashid, S.S., Maity, R.: Prediction of monthly rainfall on homogeneous monsoon regions of India based on large scale circulation patterns using genetic programming. J. Hydrol. **454–455**, 26–41 (2012)
3. Karmakar. S., Kowar, M.K.: Long-range monsoon rainfall pattern recognition and prediction for the subdivision 'EPMB' chhattisgarh using deterministic and probabilistic neural network. In: ICAPR 2009. IEEE (2009). https://doi.org/10.1109/ICAPR.2009.24. ISBN 978-0-7695-3520-3/09 $25.00
4. Nayak, M.A., Ghosh, S.: Prediction of Extreme Rainfall Event Using Weather Pattern Recognition and Support Vector Machine Classifier. Springer, Vienna (2013). https://doi.org/10.1007/s00704-013-0867-3

5. Liu, Y., Liu, L.: Rainfall feature extraction using cluster analysis and its application on displacement prediction for a cleavage-parallel landslide in the Three-Gorges Reservoir area. Nat. Hazards Earth Syst. Sci. Discuss (2016). https://doi.org/10.5194/nhess-2015-320
6. Addisu, S.. Selassie, Y.G., Fissha, G., Gedif, B.: Time Series Trend Analysis of Temperature and Rainfall in Lake Tana Sub-basin, Ethiopia. Environment Systems Research, A Springer Open Journal (2015). https://doi.org/10.1186/s40068-015-0051-0
7. Tripathi, S., Govindaraju, R.S.: Change detection in rainfall and temperature patterns over India. In: SensorKDD 2009, June 28, 2009, Paris, France. ACM (2009) ISBN 978-1-60558-668-7. $5.00
8. Adeyemo, J., Otieno, F., Ojo, O.: Analysis of temperature and rainfall trends in Vaal-Harts irrigation scheme, South Africa. Am. J. Eng. Res. (AJER) **32**, 265–269 (2014). e-ISSN: 2320-0847, p-ISSN: 2320-0936
9. https://en.wikipedia.org/wiki/2013_North_India_floods
10. http://www.livemint.com/Politics/yJs9KtfAc6hSeZ3XglstuI/IMD-to-soon-decide-on-forecasting-model.html
11. http://dataaspirant.com/2017/05/22/random-forest-algorithm-machine-learing/
12. http://www.imd.gov.in

Reducing Stage Weight Estimation Error of Slow Task Detection in MapReduce Scheduling

Utsav Upadhyay(✉) and Geeta Sikka

Department of Computer Science and Engineering,
National Institute of Technology, Jalandhar, Punjab, India
upadhyaya.utsav@gmail.com

Abstract. Hadoop architecture mainly comprises of Hadoop MapReduce and Hadoop Distributed File System (HDFS), for processing big data sets. Distributed processing has been widely used for handling large scale data sets. In the recent years, the volume of data has been increasing exponentially and the scalability of processes is growing too. This is the reason, Hadoop architecture attracted and has been adopted by many cloud computing enterprises. MapReduce is a programming model, created and utilized effectively by Google for performing computations on its large volume data sets. LATE, SAMR and ESAMR scheduling algorithm were all introduced for improving the speculative re-execution of slow tasks over Hadoop's default job scheduler. In our work, we propose the replacement of *k*-means used in ESAMR algorithm for task's stage weights estimation by multilayered, feedforward, non-linear sigmoid perceptron model of Artificial Neural Network, thus improving the efficiency of ESAMR algorithm.

Keywords: Hadoop · MapReduce · LATE scheduler · SAMR
ESAMR

1 Introduction

Development of enormous varieties of services over the Internet is leading to exponential increase in the volume of data and the number of users of Internet. For handling the growing requirements, solutions like Distributed Computing and Cloud Computing are developed. Several reasons contribute in adding programming complexity for distributed environments. MapReduce was initially created at Google by Jeffrey Dean and Sanjay Ghemawat [1], acts as the software framework for Hadoop architecture, provides a simple yet efficient platform for building applications that processing enormous amount of data, requiring parallel processing on a multi-node cluster in an efficient and fault-tolerant way with adequate reliability.

Hadoop [2], an open source project, allowing for distributed processing of large volume data on node cluster consisting of commodity computers/hardware,

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 284–291, 2018.
https://doi.org/10.1007/978-3-319-74690-6_28

making use of an easy and efficient programming model. MapReduce framework also has scalable capabilities i.e. from a server to large number of nodes, each of which offers storage space and local computational capabilities. Hadoop was originally developed by Doug Cutting [2]. Most important benefit of MapReduce framework is its ability to automatically handle failures and thus, reducing the job of application developers [1,3,4]. In cases of node failure, MapReduce has a built in mechanism for automatically re-executing the speculative task on a different node, thereby, enhancing the overall performance [5–7]. MapReduce, over the past few years has proved its worth in various applications [4,8–11].

2 Background

This section starts with the introduction of the MapReduce programming model. It forms the basis of our proposed work. We then provide introduction to the speculative re-execution mechanisms followed by Hadoop default scheduler, LATE scheduler, SAMR scheduler and ESAMR scheduler.

2.1 Concept of MapReduce

Hadoop cluster? and MapReduce cluster? are terms, used almost conversely. A Hadoop computing cluster is made of commodity computers/nodes, where one computer/nodes acts as the master node and remaining acts as slave nodes. The Hadoop node cluster for the management of its data files, is supported by Hadoop Distributed File System (HDFS) [2]. Each file is divided into multiple blocks of same size (e.g., 128 MB) and replicated copies (e.g., 4) of all blocks are stored on local disks of worker/slave nodes. A MapReduce processing completes in two phases. First is the map phase and the second is the reduce phase. When a MapReduce job arrives on a Hadoop cluster, it is partitioned into M map tasks and R reduce tasks, every map/reduce task requires processing of one or more blocks of data. When worker node receives any map task, the corresponding input data block is scanned. The node then parses out the given key/value combinations, and each combination is then passed to a predefined map function, which in simpler terms, generates intermediate key/value combinations, which after being sorted are stored to the local storage disk, and are finally partitioned to R regions by the partitioning function. The address of the generated data, which is stored on local disks are then passed to the master node as these locations are required in the reduce phase. A reduce task make use of RPCs (Remote Procedure Calls) for reading input block i.e. the data generated from M map tasks of job. Each reduce task has the responsibility of one or more partitions (region) of generated data with allocated keys. A reduce task gets its partitions of generated data from multiple worker nodes, which were executing the map tasks earlier. This phase is also known as the shuffle process, as it involves multiple communications among the worker nodes. After receiving the required generated data, a reduce task then sorts, the generated keys grouping together all existence of a key. The reduce task, finally initiates its reduce function to generate the desired output key/value pairs. The map task executes in two stages: map ($M1$) stage and sort ($M2$) stage, while the reduce task executes in three stages: shuffle stage, sort stage, and reduce stage. In a MapReduce job, the map tasks of a job are not dependent and are executed simultaneously. The reduce tasks of the job depends on the output of map tasks as the output of map task acts as the input to reduce task. Similarly, the reduce tasks of the job are independent and are executed simultaneously. Thus, the time to complete the map task is identified by the completion time of the slowest map task, while the time to complete the reduce stage i.e. Jobs completion time is determined by the time to complete the slowest reduce task.

2.2 Slow Task Detection Algorithms

Estimating progress of a task is essential for estimating the task completion time. In estimating progress of a task a quantity, named, Progress Score (PS) is required. The Progress Score is a quantity that lies between 0 and 1. The value of Progress Score of a task is 0 at the beginning, as soon as the task starts, it starts increasing and finally reaches 1 when the Map/Reduce task is completed. The Map and Reduce task have two and three stages involved respectively. The amount of time consumed by each stage is unknown and require some estimation based on the volume, type of the job and the datasets involved. For the estimation of time required by each stages, slow task detection employs some mechanism in estimating the tasks Stage Weights. The better the estimation, more accurate is the progress score calculation, leading to improving the efficiency in slow task detection.

Consider the scenario, where a job has J number of tasks, each with one task stage to be executed in map phase and three task stages to be executed in reduce phase and task current stage in reduce phase is L, the total number of key/value combinations to be evaluated by a task is N and out of them, M are evaluated successfully

$$PS = \begin{cases} 1/2 * (L - 1 + M/N) & ForMapTask \\ 1/3 * (L - 1 + M/N) & ForReduceTask \end{cases} \tag{1}$$

Hadoops default scheduler uses an assumption that a map task consumes negligible amount of time in order stage as compared to map execution phase (i.e., $M1 = 1$ and $M2 = 0$) and in case of reduce task, its assumption is that each stage will take equal time (i.e., $R1 = R2 = R3 = 1/3$). Hadoop default scheduler calculates the PS and $PSavg$ from Eqs. (1) and (2) respectively, and then start detecting slow tasks using Eq. (3). If Eq. (3) holds for any task T_j, then T_j needs a backup task.

$$PS_{avg.} = \sum_{i=1}^{J} PS[i]/J \tag{2}$$

$$ForTaskT_i : PS[i] < PS_{avg.} - 0.20 \tag{3}$$

This approach had several drawbacks, suggested by Chen et al. [6]. Firstly, use of static stage weights by fixing the value of $M1, M2, R1, R2$, and $R3$ at 1, 0, 1/3, 1/3, and 1/3 respectively. However, the values of $M1, M2, R1, R2$, and $R3$ especially considering heterogeneous environment, where tasks run on nodes with different hardware configurations. Secondly, the scheduler launch backup tasks based on Eq. (3), which in several cases creates problems, for e.g. considering the scenario of heterogeneous environments, a low value of Progress Score doesnt necessarily suggest more execution time. For example, the shuffle phase of reduce task, is in general a bit slow as compared to the sort phase as a lot of interactions with multiple map tasks is involved. Thus, a reduce task progressing in shuffle phase might not actually be slow. However, backups are also launched in such cases since their PS, as computed by the fore-mentioned method would be lesser than 1/3.

For launching the backups of tasks with largest remaining execution time, Longest Approximate Time to End (LATE) [5] MapReduce scheduling algorithm tried to overcome the second problem of the above-mentioned re-execution strategy, but, it also used fixed stage weights. LATE also used Eq. (1) to calculation of the Progress Score (PS). Furthermore, it computes the Progress Rate (PR) and the remaining job execution time for that task (denoted by TimeToEnd) using Eqs. (4) and (5) respectively.

$$PR = PS/T_r \tag{4}$$

$$TimeToEnd = (1 - PS)/PR \tag{5}$$

Here T_r denotes the time, task T has executed. Despite the fact, LATE uses a comparatively better strategy in launching backups for the slow tasks; many times, it still chooses to re-execute faster tasks frequently. This problem is due to the fixed stage weights values leading to incorrect TimeToEnd estimation.

Self-Adaptive MapReduce (SAMR) [6] scheduler also tries to identify the speculative tasks by approximating the task execution time. SAMR, unlike LATE, uses dynamic stage weights for map tasks and reduce tasks. It keeps previous information about the stage weights on individual nodes and add it regularly after every task execution on that node. While approximating the completion time of a task executing on some particular node, SAMR uses the previously stored stage weight data on node so as to dynamically allocate the stage weights for a task. The strategy adopted by SAMR gives much better results than Hadoop default and LATE schedulers, especially considering the job execution of variety of workloads in heterogeneous environments.

ESAMR [7], an improvement over SAMR, classifies the previously stored information on each node into k clusters using k-means clustering procedure. In initial phase, when no job has finished any map task on any node, the average obtained from k clusters stage weights is utilized for job on that node. When executing job has finished any map tasks on a node, ESAMR uses the temporary map phase weight (i.e., $M1$) in accordance with the map tasks finished on that particular node. The temporary $M1$ weight is used to identify the cluster with the closest $M1$ weight. After the cluster is identified, ESAMR then make use of stage weights of cluster to which the job now belongs, to approximate the jobs completion time on that particular node and thus has better progress score calculation in the process of identification of slow tasks needed to be re-executed. ESAMR also carries out similar procedure during the reduce stage. Finally, when a job has completed, ESAMR computes the jobs actual stage weights and stores the weights as historical information for later use. Finally, ESAMR make use of k-means clustering algorithm to re-classify previous stage weight information stored on each node and stores the re-calculated average weights for all k clusters. Thus, ESAMR receives more accurate Progress Score, as it utilized efficient and comparatively more correct stage weights in estimation of completion time of the running tasks. ESAMR also has it's own slow task detection procedure for identification of slow/speculative tasks more accurately and provides better result than SAMR, LATE, and Hadoop default scheduling algorithms.

ESAMR uses k-means clustering for tasks Stage Weight Estimation. In this paper, we propose replacement of k-means clustering due to several reasons. The assumptions that k-means uses, like, there exists only and exactly k clusters, minimizing the SSE (Error Sum of Squares) is the sole and correct objective, all clusters have almost same SSE, all variables holds the equal importance for every clusters, results in several drawbacks. In the quest of minimize the within-cluster SSE, the k-means algorithm gives more importance to larger clusters. Thus, small cluster ends up far away from any centre and are basically lost. On the other hand, ANN performs far better than k-means and has proven its worth in various applications.

We, in our work have proposed the using multilayered, feedforward, non-linear sigmoid perceptron model of Artificial Neural Network in estimation of the task stage weights instead of the earlier used k-means approach to further enhance the efficiency of ESAMR.

3 Evaluation Environment and Parameters

For evaluating the performance of our work, we have used Hadoop 2.7.4 in an environment consisting of 16 machines. One node acts the master node while the remaining acts as data/worker nodes (Table 1). We compared our proposed work with the existing k-means mechanism for Task Stage Weight estimation of ESAMR algorithm designed to work in heterogeneous environments. ESAMR algorithm has already shown better results than SAMR and LATE schedulers and that is why, the comparison with mechanisms employed in LATE and SAMR are skipped. We ran multiple WordCount and Sort jobs, generally considered as classical examples of Hadoop applications for evaluation of our proposed work. Tasks stage weight estimation error is used as the measure for comparison of the performance of algorithm.

Table 1. Evaluation Environment

Nodes	Quantity	Hardware specification
Master node	1	Core i7, 2.4 GHz CPUs, 8 GB RAM, 1 Gbps Ethernet
Data nodes A	10	Core i5, 2.4 GHz CPUs, 4 GB RAM, 1 Gbps Ethernet, 2 map and 2 reduce slots per node
Data nodes B	5	Core i3, 2.2 GHz CPUs, 2 GB RAM, 100 Mbps Ethernet, 2 map and 1 reduce slots per node

3.1 Setting Parameters of Our Algorithm

We have used multilayered, feedforward, non-linear sigmoid perceptron model of Artificial Neural Network. We used this model as it is suitable for function approximation. The learning of the model is supervised and the learning algorithm used is gradient descent. Table 2 gives the value of parameters required in training the feed-forward neural network.

Table 2. Parameters for training feed-forward neural network

Parameter name	Value	Description
NN_{il}	4	Number of neurons input layer
N_{hl}	3	Number of hidden layers
NN_{hl}	5	Number of neurons in one hidden layer
NN_{olm}	2	Number of neurons output layer(Map task)
NN_{olr}	3	Number of neurons output layer(Reduce task)
Epochs	2000	Maximum number of epochs to train
Goal	0	Performance goal
Lr	0.9	Learning rate
Max fail	50	Maximum validation failures
Mc	0.95	Momentum constant
Min grad	$1e^{-10}$	Minimum performance gradient

4 Experimental Results

To compare the estimates of proposed work and existing k-means mechanism employed in ESAMR, we have used average relative error as the measure. Calculation of the same shows reduction in the relative error in our proposed work. Relative error is calculated by using Eq. (6). Figures 1 and 2 shows the errors in task stage weight estimation of proposed work and ESAMR algorithm on map and reduce task of WordCount job. The figures clearly indicate the reduction in the tasks stage weight estimation error.

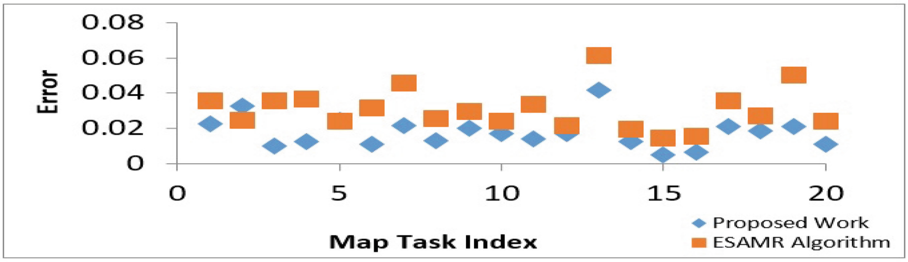


Fig. 1. Error in task stage weight estimations Vs Map task Index (Word count problem).

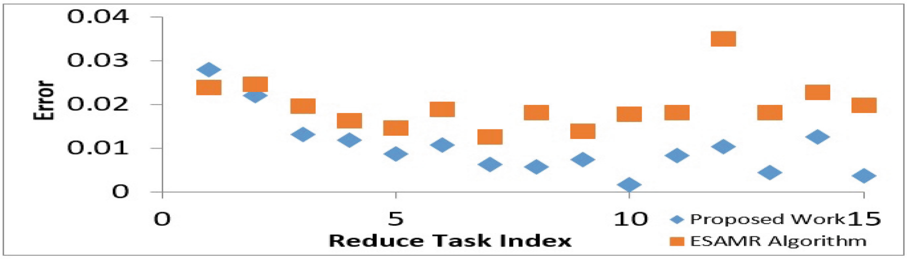


Fig. 2. Error in task stage weight estimations Vs Reduce task Index (Word count problem).

$$RelativeError(in\%) = (|Real - Estimated|/Real) * 100 \tag{6}$$

Figure 3 shows the relative error of proposed algorithm and ESAMR algorithm on WordCount job only. The figure clearly indicates the reduction in the tasks stage weight estimation error. This reduction in tasks stage weight estimation error improves the process of progress score calculation and the overall efficiency of the ESAMR scheduling algorithm.

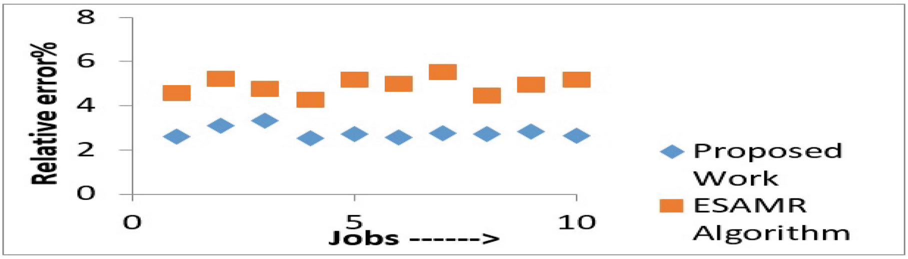


Fig. 3. Average relative error of task stage weight estimations Vs WordCount jobs.

The picture depicted in the figures represents a part of the results obtained and the results shown for WordCound jobs depict similar picture in Sort jobs. Over a large number of experiments conducted, the proposed replacement of k-means by Artificial Neural Network will reduce the task stage weight estimation error by 8–10% and improves the overall efficiency of Slow Task Detection of ESAMR algorithm.

5 Conclusion and Future Scope

We have successfully improved MapReduce re-execution mechanisms in terms of tasks stage weight estimation error leading to overall improvement in the performance of ESAMR algorithm. Experimental results have clearly indicated the effectiveness of proposed work. We further plan to develop a much more efficient slow task detection procedure with backup launching strategy to further improve the efficiency in slow task detection algorithms.

References

1. Dean, J., Ghemawat, S.: Mapreduce: simplified data processing on large clusters. Commun. ACM **51**(1), 107–113 (2008)
2. Welcome to apache hadoop!. http://hadoop.apache.org/. Accessed on 10 Oct 2017
3. Tran, H.M., Schönwälder, J.: Distributed case-based reasoning for fault management. In: AIMS, pp. 200–203. Springer, Heidelberg (2007)
4. Jin, C., Buyya, R.: MapReduce programming model for. net-based cloud computing. In: European Conference on Parallel Processing, pp. 417–428. Springer, Heidelberg (2009)
5. Zaharia, M., Konwinski, A., Joseph, A.D., Katz, R.H., Stoica, I.: Improving MapReduce performance in heterogeneous environments. In: Proceedings of the 8th USENIX conference on Operating systems design and implementation, OSDI 2008, pp. 29–42 (2008)
6. Chen, Q., Zhang, D., Guo, M., Deng, Q., Guo, S.: SAMR: A self-adaptive MapReduce scheduling algorithm in heterogeneous environment. In: 2010 IEEE 10th International Conference on Computer and Information Technology (CIT), pp. 2736–2743. IEEE (2010)
7. Sun, X., He, C., Lu, Y.: ESAMR: An enhanced self-adaptive MapReduce scheduling algorithm. In: 2012 IEEE 18th International Conference on Parallel and Distributed Systems (ICPADS), pp. 148–155. IEEE (2012)
8. Zhang, S., Han, J., Liu, Z., Wang, K., Feng, S.: Spatial queries evaluation with MapReduce. In: Eighth International Conference on Grid and Cooperative Computing, GCC 2009, pp. 287–292. IEEE (2009)
9. Marozzo, F., Talia, D., Trunfio, P.: Implementing MapReduce applications in dynamic cloud environments. In: Cloud Computing, pp. 211–223. Springer, Cham (2017)
10. Amato, F., Moscato, F.: Model transformations of MapReduce design patterns for automatic development and verification. J. Parallel Distrib. Comput. **110**, 52–59 (2016)
11. Rose, K., Baker, V., Bauer, J., Vasylkivska, V.: Innovating big data computing Geoprocessing for analysis of engineered-natural systems. In: AGU Fall Meeting Abstracts (2016)

Analysis of Complete-Link Clustering for Identifying Multi-attributes Software Quality Data

Jaya Pal[✉] and Vandana Bhattacherjee

Department of CS and Engineering, Birla Institute of Technology,
Mesra, Ranchi, India
{jayapal,vbhattacharya}@bitmesra.ac.in

Abstract. Clustering is a robust technique in the area of data mining research for extracting useful information from a set of data. It classifies the data into several clusters based on similarity of the pattern. The quality of clustering can be presented based on a metric of dissimilarity of objects, compared for various types of data. This paper presents one of the agglomerative approaches of hierarchical clustering techniques i.e. complete–link clustering by considering Euclidian distance metric for the quality estimation for students' projects data.

Keywords: Clustering · Hierarchical clustering · Data mining
Agglomerative · Complete link clustering · Cluster generation
Software quality

1 Introduction

Data mining can be viewed as a result of the natural growth of information technology [1]. Currently data mining [2, 3] is known as powerful technique and can be used in various field such as graphical presentation of database technology, machine learning, pattern recognition, information retrieval, artificial intelligence, data visualization high-performance computing, software fault prediction [13].

Clustering [4] is a powerful data mining technique which groups the data into several clusters based on similarity of the pattern. It is one of the most frequently and widely used data analysis techniques. It is used to identify natural grouping of case based on a set of attributes. Cases within the same group have more or less similar attributes values. Cluster analysis is a multivariate analysis technique where individual data points with similar characteristics are determined and grouped and dissimilar data points fall in different groups.

A set of input data points and a standard of measuring the connection between two data points is the source of the cluster analytical system. Data set of several groups is the output of the cluster analytical system. These groups form a partition. The comprehensive description of each cluster is an additional result of cluster analysis. This additional result is mainly important for analysing the characteristics of collected data. There are different methods in cluster analysis such as hierarchical clustering [9], fuzzy clustering [10], dynamic cluster etc.

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 292–300, 2018.
https://doi.org/10.1007/978-3-319-74690-6_29

In this paper, we have focused on hierarchical clustering which is essentially to combine the data points into clusters one by one. Hierarchical clustering creates sets of clusters which can be graphically represented by a tree structure, known as *dendrogram*. It is used to demonstrate the hierarchical clustering technique and the sets of different clusters. All elements are together in one cluster is the root in the dendrogram. Each leaves in the dendrogram contains single element cluster. Internal nodes in the dendrogram represent new clusters. These clusters are formed by integrating the clusters that show as children in the tree. To merge the clusters, each level in the tree is linked with the distance measure. The off springs clusters have a distance between them less than the distance value connected with this level in the tree hence all clusters formed at a particular level is merged. In this way we can obtain a specified number of clusters by cutting the dendrogram at some level [12]. Due to its nested structure it provides better structural information and is effective. Software Quality Estimation has been identified as one of the major challenges for computer science [11]. For this, three quality metrics have been gathered for 10 projects from 50 students working in 10 groups. These metrics are Graphics User Interface (GUI), Meaningful Error Message (MEM) and User Manual (UM). The rest of the paper is organized as follows: Sect. 2 presents the hierarchical clustering techniques, Sect. 3 presents illustration and analysis, Sect. 4 presents conclusion.

2 Hierarchical Clustering

Hierarchical clustering techniques are used to produce an embedded sequence of partitions [5, 6]. The output of Hierarchical clustering algorithm is represented graphically by a tree structure known as Dendrogram [3, 7].

2.1 Working Strategy

Following are two basic strategies to generate hierarchical clustering.

(a) *Agglomerative*: It is a bottom up approach. Most of the hierarchical clustering approaches belong to this strategy. It consists of the following steps:
Step 1: Each object is considered in its own cluster.
Step 2: At each step, it merges the closest pair of clusters with cluster similarity or the distance between the larger and larger cluster.
Step 3: Repeat Step 2, until all the objects are placed into a single cluster.
Step 4: Stop.

(b) *Divisive*: It is a top-down approach. It is the opposite of agglomerative approach. In which we have to decide which cluster to be splited and how to split. It consists of the following steps:
Step 1: Start with all objects in one cluster.
Step 2: At each step, split each cluster into the smaller and smaller clusters.
Step 3: Repeat Step 2, until each object makes individual cluster for itself.
Step 4: Stop

The agglomerative technique assumes that a set of elements and distance between them is given as input. Each element is considered as individual cluster. We use N × N vertex adjacency matrix, A as input. Here adjacency matrix A contains a distance value between the tuples $\mathrm{t_i}$ and $\mathrm{t_j}$, as $\mathrm{A[i,j]} = \mathrm{dis}\left(\mathrm{t_i},\ \mathrm{t_j}\right)$.

The distance between the tuples can be calculated by the following distance measure.

Euclidian distance:

$$dis(t_i,\ t_j)\ =\ \sqrt{\sum_{h=1}^{k} (t_{ih}\ -\ t_{jh})^2}$$

Manhattan distance:

$$dis(t_i,\ t_j)\ =\ \sum_{h=1}^{k} \left|t_{ih}\ -\ t_{jh}\right|$$

The clusters are allocated with sequential number r from $0, 1, 2, \ldots.., (\mathrm{N}-1)$ where N is number of objects and L(k) represents the level of k$^{\mathrm{th}}$ clustering. The distance between clusters $\mathrm{K_i}$ and Kj is denoted as $\mathrm{dis(K_i, Kj)}$.

The most well-known agglomerative techniques are as follows:

Single link technique: It uses smallest distance minimum distance between an object in one cluster and an object in the other cluster.

The algorithm is composed of the following steps:

Step 1: Start with the disjoint clustering having level $\mathrm{L(0)} = 0$ and sequence number m = 0.

Step 2: Find the least disjoint pair of clusters $\mathrm{K_i}$ and Kj such that $\mathrm{dis(K_i, Kj)} = \mathrm{min}\left[\mathrm{dis}\left(\mathrm{t_{il}, t_{jm}}\right)\right]$, $\forall\ t_{il}\ \in\ K_i\ \notin\ K_j$ and $\forall\ t_{jm}\ \in\ K_j\ \notin\ K_i$ where minimum is consider from all pairs of clusters in the current clustering.

Step 3: Increment the sequence number $\mathrm{r} = \mathrm{r} + 1$. Merge clusters $\mathrm{K_i}$ and Kj into single cluster to form next clustering r. Set the level of clustering as $\mathrm{L(r)} = \mathrm{dis(K_i, Kj)}$.

Step 4: Update the adjacency matrix A by deleting the rows and columns corresponding to clusters $\mathrm{K_i}$ and Kj and add new row and column corresponding to newly formed cluster. The distance between new cluster denoted as $\mathrm{(K_i, Kj)}$ and old cluster as C is defined as $\mathrm{dis(C, (K_i, Kj))} = \mathrm{min[dis(C, K_i)}$, $\mathrm{dis(C, Kj)]}$.

Step 5: If all objects are in one cluster then stop otherwise go to Step 2.

Average link technique: It uses average distance between an object in one cluster and an object in the other cluster.

The algorithm is composed of the following steps:

Step 1: Start with the disjoint clustering having level $L(0) = 0$ and sequence number m = 0.

Step 2: Find the least disjoint pair of clusters K_i and Kj such that $dis(K_i, Kj) = \min[dis(t_{il}, t_{jm})]$, $\forall t_{il} \in K_i \notin K_j$ and $\forall t_{jm} \in K_j \notin K_i$ where minimum is consider from all pairs of clusters in the current clustering.

Step 3: Increment the sequence number $r = r + 1$. Merge clusters K_i and Kj into single cluster to form next clustering r. Set the level of clustering as $L(r) = dis(K_i, Kj)$.

Step 4: Update the adjacency matrix A by deleting the rows and columns corresponding to clusters K_i and Kj and add new row and column corresponding to newly formed cluster. The distance between new cluster denoted as (K_i, Kj) and old cluster as C is defined as $dis(C, (K_i, Kj)) = mean[dis(C, K_i), dis(C, Kj)]$.

Step 5: If all objects are in one cluster then stop otherwise go to Step 2.

Complete link technique: It uses largest distance between an object in one cluster and an object in the other cluster.

The algorithm is composed of the following steps:

Step 1: Start with the disjoint clustering having level $L(0) = 0$ and sequence number m=0.

Step 2: Find the least disjoint pair of clusters K_i and Kj such that $dis(K_i, Kj) = \min[dis(t_{il}, t_{jm})]$, $\forall t_{il} \in K_i \notin K_j$ and $\forall t_{jm} \in K_j \notin K_i$ where minimum is consider from all pairs of clusters in the current clustering.

Step 3: Increment the sequence number $r = r + 1$. Merge clusters K_i and Kj into single cluster to form next clustering r. Set the level of clustering as $L(r) = dis(K_i, Kj)$.

Step 4: Update the adjacency matrix A by deleting the rows and columns corresponding to clusters K_i and Kj and add new row and column corresponding to newly formed cluster. The distance between new cluster denoted as (K_i, Kj) and old cluster as C is defined as $dis(C, (K_i, Kj)) = max[dis(C,K_i), dis(C, Kj)]$.

Step 5: If all objects are in one cluster then stop otherwise go to Step 2.

3 Illustration and Analysis

In this paper we are considering agglomerative strategy of hierarchical clustering using complete link technique to classify the software projects on the basis of software quality and compare both the methods. Objects collection consists of 10 projects: $U = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}\}$ as shown in Table 1.

Table 1. Projects and metrics: Graphical user interface (GUI), Meaningful Error Message (MEM), User Manual (UM), Software Quality (SQ) (ranks).

Project	GUI	MEM	UM	SQ
P1	0	0.5	9	75
P2	5	0.5	14	80
P3	1	0.4	8	72
P4	7	0.7	12	82
P5	7	0.7	16	82
P6	6	0.6	14	83
P7	7	0.8	18	91
P8	1	0.2	9	62
P9	7	0.5	14	82
P10	8	0.8	17	92
Statistical analysis of above data				
	GUI	MEM	UM	SQ
Min	0	0.2	8	62
Max	8	0.8	18	92
Max-Min	8	0.6	10	30

The extracted data above is utilized to realize cluster analysis. Each project is considered as one cluster.

Metrics Used

This paper focuses on quality of software using clustering and metrics were designed and/or adapted from Pal and Bhattacherjee [8] where the authors have developed a Fuzzy Logic System for prediction of software quality.

Description of metrics:

(1) *GUI (Graphical User Interface)*: GUI was measured as the relative number of forms which were clearly displayed, on a scale of 0–10.
(2) *MEM (Meaningful Error Message):* MEM was measured as the relative number of meaningful error messages displayed by the software, on a scale of 0–1.
(3) *UM (User Manual):* UM was measured as the completeness of the user manual or help file, on a scale of 1–20.

The usability of the ultimate product (program) has been judged by team of three experts who ranked the various projects on a scale of 50–100 for usability and this served as the predicted output.

Description

Step 1: *Standardize transformation of original data.*

Each project data point as shown in Table 1 is treated as one point of 3-Dimensional Euclidean space, and 10 projects are viewed as 10 points of 3-Dimensional Euclidean space, which forms 10 * 3 matrix.

$$A = \begin{pmatrix} 0 & 5 & 1 & 7 & 7 & 6 & 7 & 1 & 7 & 8 \\ 0.5 & 0.5 & 0.4 & 0.7 & 0.7 & 0.6 & 0.8 & 0.2 & 0.5 & 0.8 \\ 9 & 14 & 8 & 12 & 16 & 14 & 18 & 9 & 14 & 17 \end{pmatrix}^{T}$$

Ten projects could be represented as 10 clusters that is $C1 = (0, 0.5, 9)$, $C2 = (5, 0.5, 14)$, $C3 = (1, 0.4, 8)$, $C4 = (7, 0.7, 12)$, $C5 = (7, 0.7, 16)$, $C6 = (6, 0.6, 14)$, $C7 = (7, 0.8, 18)$, $C8 = (1, 0.2, 9)$, $C9 = (7, 0.5, 14)$, $C10 = (8, 0.8, 17)$.

Standardize the matrix and calculate $\overline{x_j} = \frac{1}{n}\sum_{i=1}^{n} x_{ij}$

That is $\overline{X_1} = 4.9$, $\overline{X_2} = 0.57$, $\overline{X}_3 = 13.1$

From $x'_{ij} = \frac{x_{ij}}{\overline{x_j}}$, we get standardized matrix as

$$\begin{pmatrix} 0.0 & 0.87 & 0.68 \\ 1.02 & 0.87 & 1.07 \\ 0.20 & 0.70 & 0.61 \\ 1.42 & 1.23 & 0.92 \\ 1.42 & 1.23 & 1.22 \\ 1.22 & 1.05 & 1.06 \\ 1.42 & 1.40 & 1.37 \\ 0.20 & 0.35 & 0.69 \\ 1.42 & 0.85 & 1.07 \\ 1.63 & 1.40 & 1.29 \end{pmatrix}$$

Step 2: *Construction of shortest distance matrix and clustering.*
Using Euclidian distance construct the shortest distance matrix as follows:

$$A^{(1)} = \begin{pmatrix} 0 & 1.41 & 0.44 & 2.02 & 2.32 & 1.78 & 2.64 & 0.73 & 1.83 & 2.77 \\ 1.41 & 0 & 1.45 & 0.91 & 0.91 & 0.39 & 1.23 & 1.72 & 0.42 & 1.36 \\ 0.44 & 1.45 & 0 & 2.06 & 2.36 & 1.82 & 2.68 & 0.43 & 1.83 & 2.81 \\ 2.02 & 0.91 & 2.06 & 0 & 0.30 & 0.52 & 0.62 & 2.33 & 0.53 & 0.75 \\ 2.32 & 0.91 & 2.36 & 0.30 & 0 & 0.54 & 0.32 & 2.63 & 0.53 & 0.45 \\ 1.78 & 0.39 & 1.82 & 0.52 & 0.54 & 0 & 0.86 & 2.09 & 0.41 & 0.99 \\ 2.64 & 1.23 & 2.68 & 0.62 & 0.32 & 0.86 & 0 & 2.95 & 0.75 & 0.29 \\ 0.73 & 1.72 & 0.43 & 2.33 & 2.63 & 2.09 & 2.95 & 0 & 2.10 & 3.08 \\ 1.83 & 0.42 & 1.83 & 0.53 & 0.53 & 0.41 & 0.75 & 2.10 & 0 & 0.98 \\ 2.77 & 1.36 & 2.81 & 0.75 & 0.45 & 0.99 & 0.29 & 3.08 & 0.98 & 0 \end{pmatrix}$$

In the matrix $A^{(1)}$, the shortest distance is $d^{7,10} = 0.29$ and the level of aggregation is 0.29. By merging clusters C7 and C10 into new cluster C11, we get nine sub-clusters $\{C11, C1, C2, C3, C4, C5, C6, C8, C9\}$. The clustering process will continue until

$$A^{(9)} = \begin{pmatrix} 0 & 3.08 \\ 3.08 & 0 \end{pmatrix}$$

Table 2. Clustering order

Combined order	Combined clusters	Level of aggregation
1	$C11 = \{C7, C10\}$	0.29
2	$C12 = \{C4, C5\}$	0.30
3	$C13 = \{C2, C6\}$	0.39
4	$C14 = \{C13, C9\}$	0.42
5	$C15 = \{C3, C8\}$	0.43
6	$C16 = \{C15, C1\}$	0.73
7	$C17 = \{C11, C12\}$	0.75
8	$C18 = \{C17, C14\}$	1.36
9	$C19 = \{C16, C18\}$	3.08

According to $A^{(9)}$, the shortest distance is $d^{1,2} = 3.08$. By merging clusters C18 and C16 we get new clusters C19 contains all data points. The order of clustering is shown in Table 2.

Step 3: *Obtain hierarchical clustering (Complete-link) diagram.*

Generated Dendrogram using Euclidian Distance is shown in Fig. 1.

Analysis

For the hierarchical clustering analysis, it is noted from matrix $A^{(1)}$ that the shortest distance is 0.29 hence data points 7 and 10 are merged into cluster C11. The next shortest distance obtained is 0.30 and data points 4 and 5 get merged to get C12.

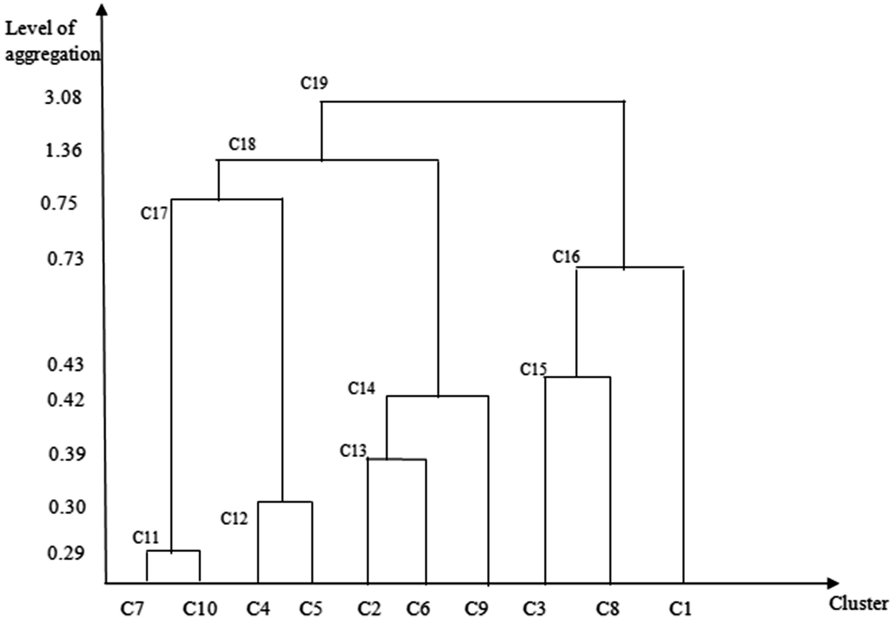


Fig. 1. Dendrogram for hierarchical clustering.

Proceeding in this manner, C11 and C12 get merged to get C17(7, 10, 4 and 5) at 0.75, data points2 and 6 get merged to get C13 at 0.39, data point 9 gets merged with C13 to get C14(2, 6 and 9) at 0.42, data points 3 and 8 get merged to get C15 at 0.43, data point 1 merges with C15 to get C16(3,8 and 1) at 0.73, C14 and C17 merge together to get C18(7, 10, 4, 5, 2, 6 and 9) at 1.36 and at 3.08 C18 and C16 merge to obtain one final cluster C19 containing all data points. The dendrogram for hierarchical clustering analysis (using complete link) is shown in Fig. 1.

4 Conclusion

This paper presents an agglomerative approach of hierarchical clustering technique, the complete link clustering. This technique proceeds as follows: First the data is transformed to a standardized form by dividing with the mean value for the attribute. Then, the Euclidean distance is used to construct the shortest distance matrix $A^{(1)}$. In the matrix, the shortest distance value is taken and the data points are merged into a cluster. This method proceeds until all the points are clustered together. In this paper the quality estimation for students' projects data based on complete link clustering as described above has been done. In this method, the distance between one pair of closest cluster is maximum to another pair of closest cluster. That means the distance between two clusters is derived by the most distant nodes in the two clusters. Due to this reason at each step the complete link clustering algorithm inclines to minimize the increase in diameter of the clusters.

We shall now analyze the clustering process further based on the threshold distance. At level of aggregation 0.73, 4 clusters are obtained the constituent projects being as follows: $\{P7, P10\}$, $\{P4, P5\}$, $\{P2, P6, P9\}$ and $\{P1, P3, P8\}$. However at level 0.75, 3 clusters are obtained as: $\{P4, P5, P7, P10\}$, $\{P2, P6, P9\}$ and $\{P1, P3, P8\}$. Upon analysis of project data, it is observed that indeed projects grouped together at level 0.75 are similar. For example consider the first cluster of $\{P4, P5, P7, P10\}$ the maximum variation in GUI values is 1 and in MEM the values is 0.1. Similar observations can be made for the other two clusters. This observation proves our point that when clusters are closely packed or compressed then the method will produce high quality clusters.

References

1. Gediminas, A., Jesse, B.: *C-TREND:* temporal cluster graphs for identifying and visualizing trends in multiattribute transactional data. IEEE Trans. Knowl. Data Eng. **20**(6), 721–733 (2008)
2. Chen, R.-S., Wu, R.-C., Chen, J.Y.: Data mining application in customer relationship management of credit card business. In: Proceedings of the 29th Annual International Computer Software and Applications Conference, Taiwan, pp. 1–2 (2005)
3. Han, J., Kamber, M.: Data Mining Concepts and Techniques. Elsevier Inc., Waltham (2006)
4. Bing, L.: Web Data Mining. Springer International Edition
5. Peng, Y., Ma, Y., Shen, H.: Clustering belief functions using agglomerative algorithm. In: Information Engineering and Computer Science (ICIECS), pp. 1–4 (2010)

6. Takumi, S., Miyamoto, S.: Top-down vs bottom-up methods of linkage for asymmetric agglomerative hierarchical clustering, granular computing (GrC), pp. 459–464 (2012)
7. Jena, A.P., Naidu, A.: Analysis of complete-link clustering for identifying and visualizing multiattribute transactional data. Int. J. Comput. Sci. Eng. Technol. (IJCSET) **4**, 850–854 (2013)
8. Pal, J., Bhattacherjee, V.: A fuzzy logic system for software quality estimation. In: Proceedings of ICIT 2009, pp. 183–187 (2009)
9. Pal, J., Bhattacherjee, V.: Hierarchical cluster generation for software quality: a comparative approach. Int. J. Eng. Technol. (IJET) **6**(4), 1827–1839 (2014)
10. Pal, J., Bhattacherjee, V.: Application of fuzzy clustering on software quality using max-min method. In: Proceedings of International Conference on Communication, Computing & Security (ICCCS). NIT, Rourkela, October 2012. Procedia Technol. **6**(2012), 67–73. Elsevier Science. ISSN: 2212-0173. https://doi.org/10.1016/j.protcy.2012.10.009
11. Brooks Jr., F.P.: Three great challenges for half-century-old computer science. J. ACM **50** (1), 25–26 (2003)
12. Liu, Y., Liu, Z.: An improved hierarchical K-Means algorithm for web document clustering. In: International Conference on Computer Science and Information Technology, IEEE Conference, pp. 606–610 (2008)
13. Bishnu, P.S., Bhattacherjee, V.: Software fault prediction using quad tree based K-Means clustering algorithm. IEEE Trans. Knowl. Data Eng. **24**(6), 1146–1150 (2012)

Student Profile in E-learning Environment Based on Two-Dimensional Ontologies

Heba A. A. Al-Mamoori[1](✉), Mohamed Elemam Shehab[2], and Essam El Fakharany[3]

[1] Information System Department, Arab Academy for Science, Technology & Maritime Transport, Cairo, Egypt
Eng_heba_it@yahoo.com

[2] Arab Academy for Science, Technology & Maritime Transport, Cairo, Egypt
melemam9@gmail.com

[3] College of Computing and Information Technology, Arab Academy for Science, Technology & Maritime Transport, Cairo, Egypt
essam.elfakharany@aast.edu

Abstract. A huge relevance has been observed in ontologies as a developing technology within the E-learning. Its groundwork paved the way for ontologies improvement and systems that use ontologies in different areas; E-learning is one of these systems. In This article, a recommendation data model is proposed, which takes the student activities and behavior of the social networks to enhance his/her account. The proposed model identifies both the student activities and behavior based on the development of two-dimensional ontologies. The first dimension is the social networks and the second one is the student portal, in anther word learning management system (LMS). The proposed model utilizes data from student engagement with the E-learning management system (Build activities) and the social network (Build behavior), Facebook. The model helps students identifies students' profile enhances the learning progress and improves learning quality. The model also provides vital information for educators, equipping them with a better understanding of each student's personality. A new technique for improves user profile based on students' behavior to social media and students' activates in LMS.

Keywords: Ontology · E-learning · LMS · Ontology matching
Ontology merging

1 Introduction

E-learning one of the learning techniques depends on presenting the educational content and deliver the skills and the concepts to the student about the information technology [1]. The communication and their multimedia in a manner that gives the student the ability to interact with the contents, the teacher and the colleagues in a synchronous or asynchronous way by the time, the place and the speed that appropriate to the student's circumstances and his or her abilities, and electronically manage the educational and learning activities and its requirements by special designed systems [2]. Now the e-learning system is a very important element in the modern learning

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 301–310, 2018.
https://doi.org/10.1007/978-3-319-74690-6_30

systems [3], its present the courses and degrees to the student by using learning management system (LMS) [4].

Now, there is no special learning system that gives to each learner or student a special learning contents and learning way according to his or her abilities and prefers and saves all the student's activities and analyzing all his or her educational pre-steps that made before. By using the ontology in the e-learning which it allows creating a specific user profile for each learner this leads to design an adaptive learning system gives the learner and the teacher to reach the learning contents in a special and more personalized way for each [5]. By founding many learning styles the learner or the student will find the learning more personalized, effective and easy, plus, the teacher will have more information about the students and they prefer and their thoughts and so their personalities [6].

Learning over social media platform became very easy and entertaining [7]. The purpose of this research is to analyze and observe different users behavior while using social networks and how to use the collected data to help in enhancing students' grades for online learning systems rather than the traditional approaches used [8]. The information gained from the data analysis will improve the ability in personalization and adaptation of course content in online learning systems [9]. Student learning behavior will encourage education institutions improved and furthermore development of the organization himself. For the improved eLearning network, each institution is trying to analyze student-learning behavior in order to identify students' interaction with the website. Many techniques are used to identify user's behavior [10].

The rest of this paper is organized into five sections: initially, we focus on the introduction of paper in the first section. The second section discusses the related works and the literature review. In the third section, discuss the methodology of the proposed e-learning model system. Results and dialogues the experimental works are assumed in the fourth section, whereas the fifth section introduces the conclusion.

2 Related Works

This section reviews the relevant related literature for this study, focusing on LMS and social network. Nowadays, it is common to use LMS and social network around the world, However, Focus on LMS and social network with some A little. Algosaibi et al. [11] Proposed an approach of E-learning using three-dimensional ontology mode action matrixes that is generic scale for testing out an ontology in order to better understand and mode it, where matrix approach was used as a backbone to the searching point. Based on identifying the major pillars in building ontology, it shows how these pillars can assist in disclosing the effort of modifying ontology. Hooi et al. [12] Presented the works done through a comparative study with the goal of identifying overlaps and establishing a relationship of the various criteria. Naren et al. [13] student performance in academics using data mining and also with the ontology-based application. Feedback thereby plays an important role in analyzing the deeds a person does and helps them in an amicable way to realize their level and improvise those places where challenges are faced. Zhang et al. [14] Presented a cognitive model for ontology learning system, the used method mainly discussed six kinds of strategies for dealing with the relationship between ontology and statements: adding a statement, deleting a statement, revising with statements, updating with ontology. Jiménez-Ruiz et al. [15] proposed an approach Log Map is a highly scalable ontology matching system with 'built-in' reasoning and inconsistency repair capabilities. Gawich et al. [16] Proposed a use one or more ontology matching techniques, the use of ontology matching tools. Previous papers methodology has been reviewed and studied, which helped in understanding how the ontology is built, how it is been evaluated using onto metric, how to merge between two ontologies using two tools (string equality, Logmap). All these methodologies helped to understand the way, how a conductive ontology is built. This paper presents modification in the user profile for the student by merging between two ontologies, which are a social network, and LMS.

3 E-learning Model System

3.1 The Proposed Model

The Fig. 1 shows the e-learning in ontology and the individual components to implement our approach. The structure of the system consists the implementation depends mainly on two phases and each phase has some steps. Phases one are collection, preprocessing of data based on the Social network. And phase two is collection, preprocessing of data based on LMS.

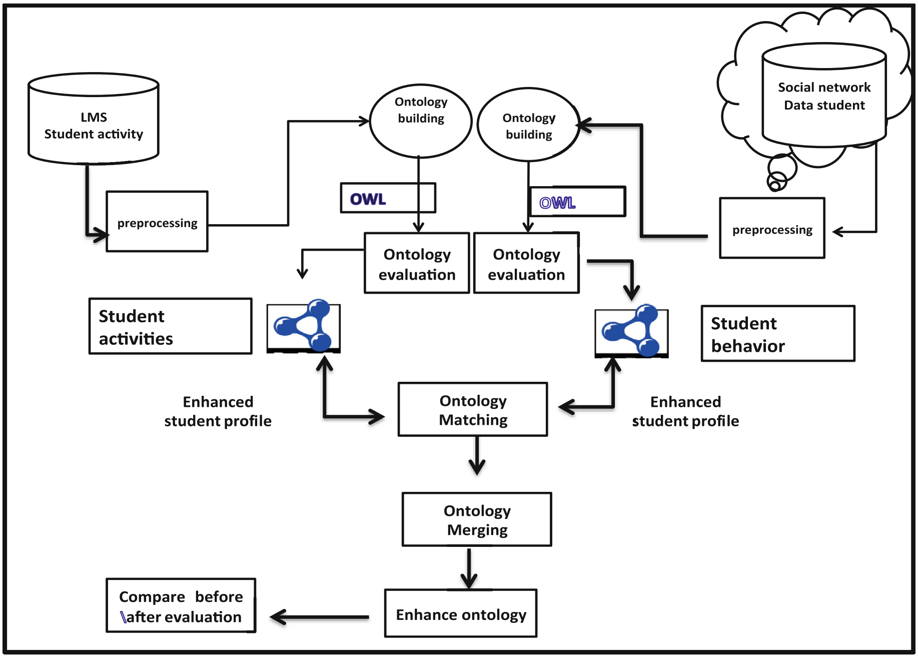


Fig. 1. Proposed model conceptual design

Due to the popularity of online social networks among college students used social networks motivating the student and change student behavior. This data can be used to create enhance profile student and student behavior type. The LMS data can be used to create enhance profile student and student activity type. The matching and merging between two ontology will help us commend the best learning object way for each student to improve the quality of teaching and learning processes in the e-learning system.

3.2 Data Collection

In this study, [17] the authors select a sample of students behavior data dataset used was collected from OU Open University from the UK in the Social network. As well as LMS dataset used was collected from OU Open University from the UK. Shown in Tables 1 and 2 some data is used. The authors use the selected students' behavior data in the Social network and student activity in LMS to improvement enhance profile student.

Table 1. Sample dataset student Attributes in LMS

Attributes	Explanation
Gender	The student's gender
Region	Identifies the geographic region
Education Level	Educational level student
Age band	The student's age
Num_of_prev_attempts	The number times the student has attempted
Studied credits	The total number of credits for the Material the student is currently studying

Table 2. Sample dataset student Attributes in Social network

Attributes	Explanation
Pid_P	Main Post id
Id_student_P	Id of the user posting
Student_Name_P	User's name
Time_Stamp_P	Time spent visiting different pages
Shares_P	Number of shares
Likes_P	Number of likes

3.3 Model Objectives

The proposed model shows how to determine students' profile using social media and LMS and how it can give more accurate determination. Design and implement an E-learning based on two-dimensional ontologies first Social networking extract the ones behaviors and the second student portal Extract the ones activity. The determination of students' profile will be dynamic which means will be identified based on his/her activates over the social media the first then LMS. By Build Social skills ontology and Build learning profile ontology matching and merging two-dimensional ontologies. Not only building ontology two-dimensional ontologies will be determined but used ontology matching and merging ontology in e-learning. The main purpose of the research is to Enhanced student profile by Student behaviour from the Social network and Student activities from LMS will be much more accurate.

3.4 Proposed Model

In this paper have a seven-step, the First step, Taken from the Internet dataset Social network Data student and LMS Data student activity. The Second step, pre-processing in pre-processing selects from data main class to Social network and LMS and select Data property, relation, etc. the Third step, ontology building used protégé program. The Fourth step, the paper took each social network only and LMS only evaluation used Onto Metric. The Fifth steps, Ontology Matching between two ontology show me new information. The Sixth step, Ontology Merging. And the last step, Ontology re-evaluation after the merging.

4 Experimental Results

Our study is based on data collected form in the UK FOR 200 Student. Student profile improvement was determined according to their behavior student in the Social network and Student activities in LMS. Some features were chosen to support the of Enhanced student profile. The evaluation results the two-dimension ontology use Ontology Metrics is a web-based tool that validates and displays statistics about a given OWL ontology, including the expressivity of the language it is written in.

4.1 Ontology Evaluation Results

Schema metrics address the design of the ontology. The study applied Ontology Metrics on the collected behavioral and active student of the Social network and LMS. The pole of each column presents special LMS and the right one present's Social network. Figure 2 shows the ratio of different metrics for the LMS ontology, that 71.11% of the attribute richness ratio, 88.89% of the inheritance richness ratio, 55.56% of the relationship richness ratio, 14.97% of the axiom/class ratio, and 57.14% of the class/relation ratio. The figure shows the ratio of different metrics for the Social network ontology. That 57.50% of the attribute richness ratio, 87.50% of the inheritance richness ratio, 50% of the relationship richness ratio, 14.55% of the axiom/class ratio and 50% of the class/relation ratio.

Knowledgebase Metrics the way data is placed within ontology is also a very important measure of ontology quality. The pole of each column presents special LMS and the right one present's Social network. Metrics in this category indicate the Average Population the average distribution of instances across all classes and Class Richness this metric is related to how instances are distributed across classes.

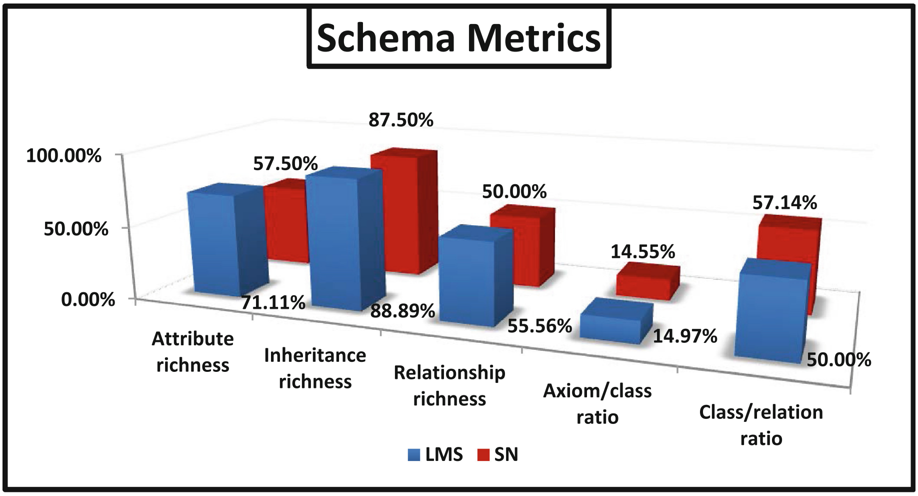


Fig. 2. Ontology schema metrics before merging

The results in Fig. 3 show the ratio in LMS, its Average population of the 17.68% and it's class richness of the 88.89%. The ratio in the social network, it's average population of the 17.70% and it's class richness of the 87.50%.

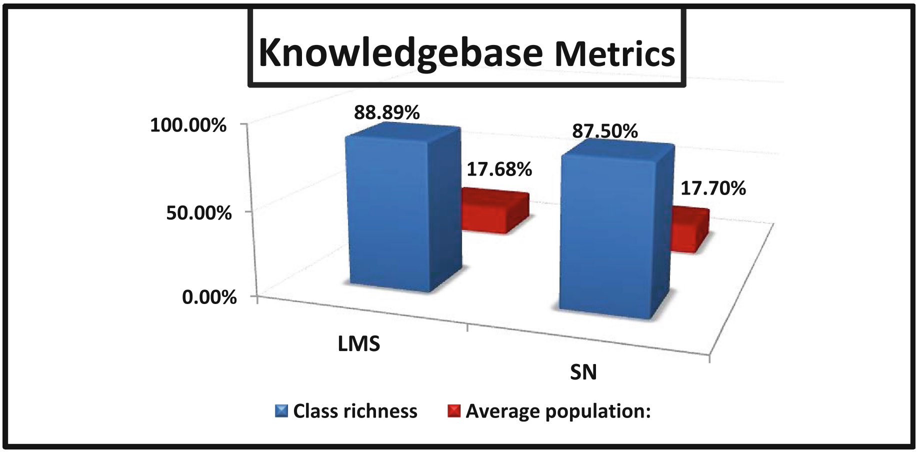


Fig. 3. Ontology knowledgebase metrics before merging

4.2 Ontology Matching Results

Ontology matching is necessary of our work, by providing a means to related concepts from different ontologies. The ontology matching method takings as input two ontologies and outputs a set of correspondences between semantically linked ontology concepts. Used in the search Ontology matching two tools. In the first tool, Log Map is a highly scalable ontology matching system with ‘built-in’ reasoning and inconsistency repair capabilities. Log Map extract mappings between classes, properties, and instances [14]. Log Map ontology matching tool to Social network and LMS ontology, Log Map detects 3 common classes and 7 data properties.

In the second tool, String Equality is the method goals to discovery similar entities (classes and subclasses) between any two ontology files in a certain domain. By the use of String Equality tool [15]. The second tool is the String Equality tool gets two matching between the two input ontologies. It detects 3 exacted matched classes and one semantic matched class. Therefore, String Equality tool shows good results than the Log Map tool and it’s suitable with version protégé 5.

4.3 Ontology Merging Results

Merging is a process of generating the creation of a new ontology from two or more existing ontologies. Its method can be implemented in a number of ways, manually and automatically. In this paper, create merge between LMS and social network class using protégé 5 by java programming language. The ontology result was a match between two classes (LMS and social network) and creates a new class to show activities and behavior for students. In Fig. 4 shown as the ontology merging results.

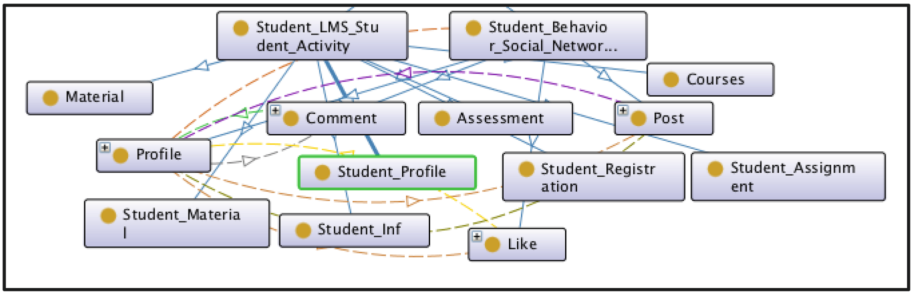


Fig. 4. Ontology merging results

4.4 Ontology Re-evaluation After the Merging

We collected the results from merging two ontology, we compared the results of the Social network only and LMS only without merging. Based on these results a merging two ontology with some improvement Students’ profile enhances. The study applied Ontology Metrics on the collected behavioral and active student of the Social network and LMS. Figure 5 shows the ratio of different metrics for the LMS ontology, that 71.11% of the attribute richness ratio, 88.89% of the inheritance richness ratio, 55.56%

of the relationship richness ratio, 14.97% of the axiom/class ratio and 50% of the class/relation ratio. And the ratio in Social Network (SN) ontology, that 57.50% of the attribute richness ratio, 87.50% of the inheritance richness ratio, 50% of the relationship richness ratio, 14.55% of the axiom/class ratio and 57.14% of the class/relation ratio. The ratio in merging ontology, that 66% of the attribute richness ratio, 80% of the inheritance richness ratio, 63.64% of the relationship richness ratio, 15.52% of the axiom/class ratio and 45.45% of the class/relation ratio.

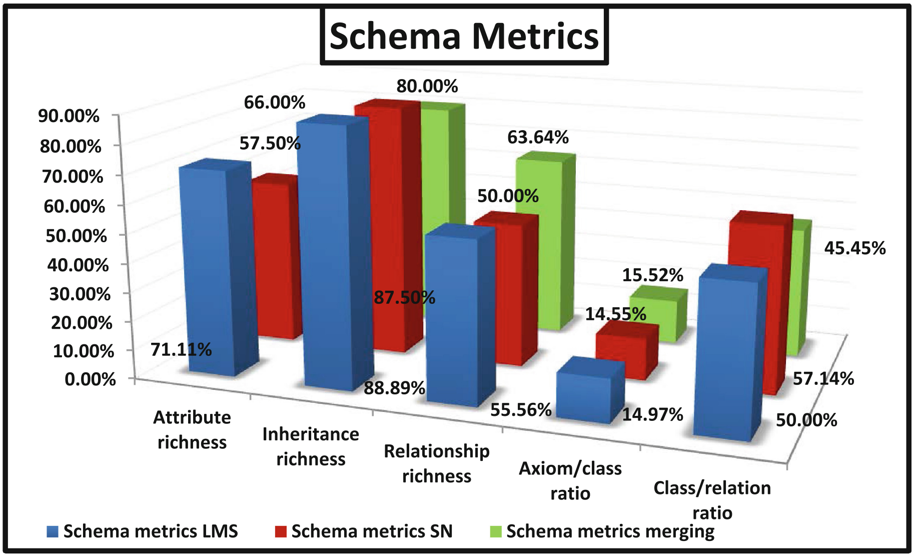


Fig. 5. Ontology schema metrics after the merging of two ontologies

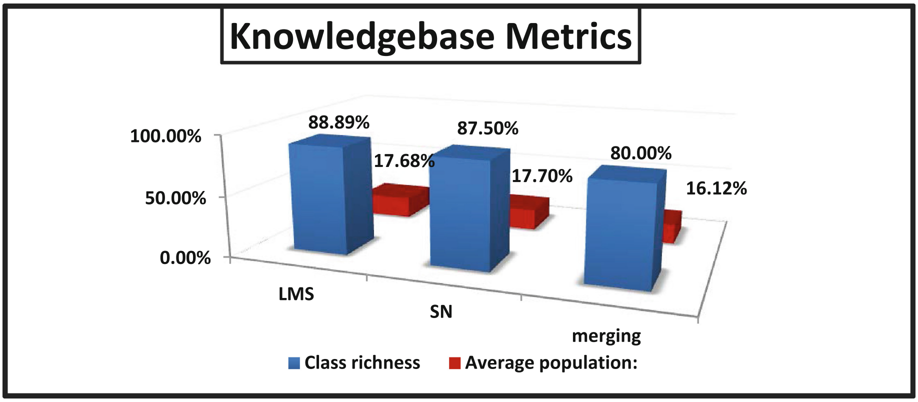


Fig. 6. Ontology knowledgebase metrics after metrics the merging of two ontologies

The results in Fig. 6 show the ratio in LMS, its Average population of the 17.68% and its Class richness of the 88.89%. The ratio in Social Network (SN), it's average population of the 17.70% and it's class richness of the 87.50%. The ratio in merging, it's average population of the 16.12% and its class richness of the 80.00%.

5 Conclusion

This paper presents the process of building the ontology structure. It starts by definition of the choice of domain e-learning and then followed by building the classes. The object properties and datatype properties also have been described in the structure. This model used a dataset social networking and the second student portal (LMS), which allowed us to have a more accurate prediction based on the activity and behavior of the student. As the paper discussed will present a hybrid ontology model based on both activity and behavior of students to enhance the user profiling process in e-learning environment using two-dimensional ontologies. The paper, which Based on two-dimensional ontologies will be evaluated by various methods:

- Building the ontology is based on the construction of two E-learning ontologies "Social networking and LMS".
- After building the ontology, Pellet reasoner is used to checking similar class by way existing simulation in value, gives a new relationship among classes with each other, recommends same individual class through recommending improve forming ontology, consistency, and classification of the ontology by checking whether a class is consistent.
- Two research models tools are proposed for ontology matching, the first tool is "string equality" technique and the second one is the "Logmap".
- Merging is the creative process that creates one ontology from two or more existing ontologies.
- Onto Metric evaluation technique, used to make the comparison between the before and after merging ontologies.

This methodology provides a more accurate evaluation and prediction of to enhancement student profile. The proposed method makes the E-learning easier to the student.

References

1. Klašnja-Milićević, A., et al.: E-Learning Systems: Intelligent Techniques for Personalization, vol. 112. Springer, Switzerland (2016)
2. Grabara, J., Bosun, P.: Consideration on online education in Romania. Int. Lett. Soc. Humanist. Sci. **14**(1), 59–65 (2014)
3. Zaharias, P., Pappas, C.: Quality management of learning management systems: a user experience perspective. Curr. Issues Emerg. eLearning **3**(1), 5 (2016)

4. Nafea, S.M., et al.: A novel adaptive learning management system using ontology. In: 2015 IEEE Seventh International Conference on Intelligent Computing and Information Systems (ICICIS). IEEE (2015)
5. Staab, S., Studer, R. (eds.): Handbook on Ontologies. Springer, Heidelberg (2010)
6. Uche, A.O., Obiora, A.V.: Social media typology, usage and effects on students of Nigerian Tertiary Institutions. Int. J. Innov. Res. Dev. **5**(8), 15–26 (2016)
7. Simões, J., Díaz Redondo, R., Fernández Vilas, A.: A social gamification framework for a K-6 learning platform. Comput. Hum. Behav. **29**(2), 345–353 (2013)
8. Muhammad, A., et al.: Learning path adaptation in online learning systems. In: 2016 IEEE 20th International Conference on Computer Supported Cooperative Work in Design (CSCWD). IEEE (2016)
9. Mary Harin Fernandez, F., Ponnusamy, R.: Ontology-based modeling student learning behaviour analysis in digital library domain knowledge using Markov chain and GUHA. In: 2015 Seventh International Conference on Advanced Computing (ICoAC). IEEE (2015)
10. Youssef, A.B., Dahmani, M., Omrani, N.: Information technologies, students'e-skills and diversity of learning process. Educ. Inf. Technol. **20**(1), 141–159 (2015)
11. Algosaibi, A.A., Melton, A.C.: Three dimensions ontology modification matrix. In: 2016 2nd International Conference on Information Management (ICIM). IEEE (2016)
12. Hooi, Y.K., Fadzil Hassan, M., Shariff, A.M.: Ontology evaluation—a criteria selection framework. In: International Symposium on Mathematical Sciences and Computing Research (iSMSC). IEEE (2015)
13. Naren, J., Ashokkumar, K., Raghavendran, V.: A semantic feedback on student's performance with data mining techniques: state of the art survey. In: 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom). IEEE (2016)
14. Zhang, D., et al.: A new cognitive model for autonomous ontology learning. In: 2016 IEEE 8th International Conference on Intelligent Systems (IS). IEEE (2016)
15. Jiménez-Ruiz, E., et al.: LogMap family results for OAEI 2014. In: Proceedings of the 9th International Conference on Ontology Matching, vol. 1317. CEUR-WS. org (2014)
16. Gawich, M., et al.: Alternative approaches for ontology matching. Int. J. Comput. Appl. **49** (18), 29–37 (2012)
17. Kuzilek, J., et al.: OU analyse: analysing at-risk students at the Open University. Learn. Anal. Rev., 1–16 (2015)

Comparative Sentiment Analysis on a Set of Movie Reviews Using Deep Learning Approach

Koyel Chakraborty[1], Siddhartha Bhattacharyya[2](✉), Rajib Bag[1], and Aboul Ella Hassanien[3,4]

[1] Department of CSE, Supreme Knowledge Foundation Group of Institutions, Mankundu, WB, India
{koyel.chakraborty,rajib.bag}@skf.edu.in
[2] Department of CA, RCC Institute of Information Technology, Kolkata, WB, India
dr.siddhartha.bhattacharyya@gmail.com
[3] Faculty of Computers & Information, Cairo University, Cairo, Egypt
[4] Scientific Research Group in Egypt, Cairo, Egypt
aboitcairo@gmail.com

Abstract. This paper provides an insight to one of the recent additions in the turf of Machine Learning culture - the process of learning representation or features, known as Deep Learning. It is highly anticipated that Deep Learning will fare much better than the traditional machine learning algorithms not only because of scalability but also of its ability to perform automatic feature extraction from raw data. This paper deals with the analyzing of sentiments on a set of movie reviews, which is considered to be the most demanding facet of NLP's. In this paper, Google's algorithm Word2Vec has been applied on a large movie review dataset to classify text so that the semantic associations between the terms stay conserved. A comparative study of the performances of some notable clustering algorithms is demonstrated concerning their application involving a variable number of features and classifier types as well as variable number of clusters.

Keywords: Machine learning · Deep learning · Sentiment analysis · Reviews Text classification · Clustering

1 Introduction

Making computers think the way we humans do or to help create an intelligence of its own is the focus of research for a long time now. Hence all the necessary information should be provided or made ready to the machine to help it imbibe the features. Various *learning algorithms* have been implemented in this regard, but the problem persists in the correct interpretation of the scenario using its intelligence. Human brain being of Deep Architecture, Deep Learning has emerged to be one of the latest areas of Machine Learning whose success story is entirely dependent on choice of appropriate data and its various representations. Machine learning is the process of taking some data, training a model using that data, and utilizing the trained model to make predictions on fresh data. An iterative predict and adjust process is continued unless the predictions no longer

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 311–318, 2018.
https://doi.org/10.1007/978-3-319-74690-6_31

yield new results. Discovery of perfect features results in the production of effective machine learning [1]. Representations of data or features should be such that it facilitates extraction of information to build appropriate predictors. Feature learning is one of the most important aspects of extracting useful patterns. If in a picture, the indicator is kept as bluish pixels for denoting sky and brownish pixels for denoting mountains, it would have two-fold advantages- one, the machine would easily be able to differentiate between the classes, and secondly, the number of classes will be measured which is required for a good classification. Efficient feature selection results in the better prediction of data. However, it has to be kept in mind that the same feature might not be applicable for all data sets. Hence, the main concern exists with developing algorithms that engineer features without human intervention.

In case of Feature Learning hierarchically, at the onset, all required features were incorporated into several non-linear feature layers and were fed to the classifier after being extorted for being easier to forecast. To create newer features we have to operate not only on the inputs but also on the initial features. But this method faced the gradient descent problem where gradients become smaller and making it unable to train architectures.

Deep Learning emerged to overcome the quandary of gradient descent so that training architectures was possible by assimilating data with innumerable non-linear hierarchical feature layers. Deep networks can be envisaged as a compilation method to discover features using several stages of uneven operations where the order of feature generation at the topmost level initiates from the lower level [2]. Deep learning should dedicate its success for having the following available-huge amounts of training data, powerful computational infrastructure, and advances in academia.

One of the major prevalent interests of society has been in public opinion. It is a trend to assimilate the mass analysis on the acceptance or rejection of a situation, product or a measure. Sentiment analysis, a task of Natural Language Processing automatically finds features conveyed by any text [3]. With the recent explosion of emerging social media trends opinions are constantly being gathered and are also followed by people for their interests through the social networks, blogs, and tweets. Companies utilize these opinions to explore and evaluate customer contentment, fondness, and product reviews.

Sentiment Analysis is the method to evaluate written or spoken the language to determine if the expression is favorable, unfavorable or neutral, and to what degree. Analysis is based on the flavour of the expression being positive, negative or neutral [4], or on the foundation of stances, i.e., pros and cons [5], or by identifying the target, i.e., whether it is a product or a measure [6], or it might be the owner of an opinion [7]. The analysis is also done on effects [8] or by determining the feature of the target that people like or dislike [9]. Depending on the application, it is to be remembered that, more refinements can be done on the analyzing criteria. Document Level, Sentence Level, Entity and Access levels are the general classifications of Sentiment Analysis techniques [3].

Studies illustrate that Sentiment Analysis is being used in a variety of applications nowadays. The application areas range from business, promotion, feedbacks, reviews and social media. Not only this, Sentiment Analysis has gained a prime importance in decision making. A survey conducted in April 2013 shows that around 90% of decisions taken on products are based on online reviews [10]. As mentioned above, polarity, [4], attitude [5], identification [6], whether being a possessor of an opinion [7] or effects [8] can easily be utilized to analyze the sentiment of a text or document. Three Sentiment Analysis types have been branded, namely, document level, sentence level and entity and aspect level as cited in [3].

Though the history behind Deep Learning algorithms applied for Sentiment Analysis dates back to 2005, in which all the base classifiers were combined to yield a better result than that a particular one, the classification being relied on the methodology in which predictions are made and learning processes did [11]. Specifically for text classification, extracting complex features from the text, building the relevant features and selecting an appropriate classification algorithm is the main ideology of machine learning driven system [12–14]. This paper travels through the process of applying Deep Learning in Sentiment Analysis.

2 Data Set Used

This paper was implemented using a labeled data set that comprises of 50,000 IMDB reviews of movies, particularly chosen to analyze sentiments. Sentiments have been expressed in the binary format, i.e., the value 0 is assigned as a sentiment score if the IMDB rating is less than 5, and one if the IMDB rating is greater than or equal to 7. The maximum number of reviews for each movie is not more than 30. It has been checked that there is no presence of any of the same movies in the 25,000 review labeled training set and the 25,000 review test set. Other than this, 50,000 IMDB reviews have also been made available which are not rated with any labels.

2.1 File Descriptions

- **labeledTrainData** – This is a tab-delimited file which contains the labeled training set. There is a header row along with 25,000 rows comprising of an id, sentiment, and text for an individual review.
- **testData** – This tab-delimited file contains the test set, whose sentiment is to be predicted. There is a header row along with 25,000 rows having an id and test for an individual review.
- **unlabeledTrainData** – This is an additional training set which is not labeled. The file is tab-delimited containing a header row along with 50,000 rows of an id associated with a text for each review.

2.2 Data Fields

- *Id is* depicting the exclusive ID of each review.
- *Sentiment is* demonstrating the review sentiment; 1 signifies a positive review and 0 a negative one.
- *Review* is to designate the Text of the review.

3 Algorithm

This paper implementation primarily deals with Mikolov et al.'s word2vec model of word representation, where vector representations of words are made to learn. Word2vec is an example of a computationally competent analytical model in which word embeddings are learned from the unrefined text. The main advantage for learning features in Word2vec model is that there is no requirement of a fully probabilistic model. We find two diverse essences of this model, (1) Continuous Bag-of-Words model (CBOW) and (2) the Skip-Gram model (Figs. 1 and 2). It is the common perception of any bag-of-words model that knowledge/feature is learning of what a word means can be easily assumed by observing the words those are inclined to it. Hence, a binary classifier is used to train the models to discriminate the real intended words from the noisy words in the similar contexts.

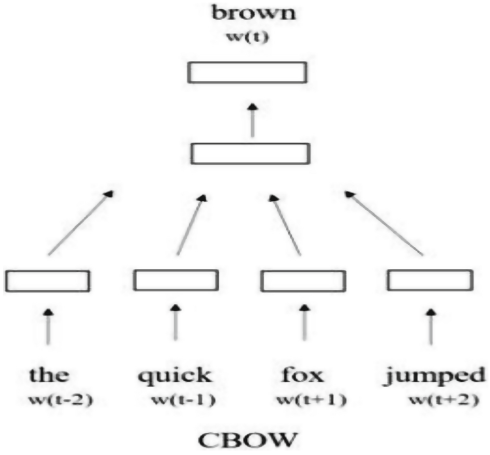


Fig. 1. *CBOW architecture* where target words are predicted from source words.

The CBOW model, on one hand, prepares each word against its perception. It queries that if a set of context words are provided, what shall be the misplaced word that is likely to be visible in the same instance. Skip-gram Model, on the contrary, trains individual contexts against a specific word. It queries that provided a single word, what are the other words that should emerge in its proximity at the same instance. CBOW model works well on syntactic representations, but training is faster; Skip-gram model works well on semantic representation, but the training is faster. These two models are conceptually very similar to the Bi-gram model but easily overcame its drawbacks of the context being very small and it including the preceding and the following words as well.

The entire method descried in this paper has been implemented in python language. Unlabeled train dataset has been taken as the input, on which data cleaning and text processing were done. This pre-processing involved removal of all the HTML tags and punctuations by "Beautiful Soup" Python Library, replacing of numbers and links by tags and finally removing stopwords. Stopwords are commonly used words that have

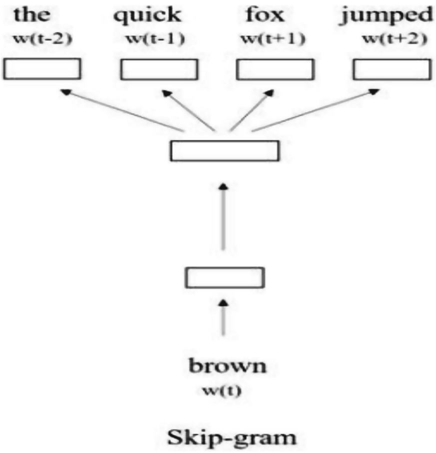


Fig. 2. *Skip gram architecture* which predicts context words from source target words.

been programmed to be ignored, as they might unnecessary take up space in our database and also consume preprocessing time. Natural Language Toolkit in python has a list of stopwords stored in 16 different languages. After this, the raw reviews are converted to a string of words. The reviews are then collected, cleaned and parsed. Ultimately the main features are mined.

Three hundred dimensional space, forty minimum words and ten words in context have been used as features to train the Word2vec model. Such demonstration seems to capture multiple linguistic regularities. The inability of Deep Learning architectures to process strings or plain text in untreated format requires the input to be numbered to carry out any job. This marks a huge advantage for the Word2vec model, whereby similar semantic words can be found out by calculating the distance between words only. A sample table is illustrated in Table 1.

Clustering has been followed in the next step. In this process, similar semantic words have been grouped. Vector Quantization is done with the built-in “cython” package [16] in which the k means algorithm has been used. The number of clusters is taken as input to the algorithm to generate k, along with a set of observation vectors to the cluster. This, in turn, returns a set of centroids, which consists of a centroid for each cluster for all k clusters. And then, each type of cluster is grouped by the cluster number or the index of the centroid closely associated with it.

As ensemble learning focuses on techniques to combine results of different trained models to produce a more accurate classifier, the random forest algorithm has been used which despite the simplicity performs excellently regarding classification. A change in the number of trees (50,100,200,500) has been applied which marginally performs better regarding timings as had been thought of to take much more time. If the number of trees along with the number of clusters can be improved, the model gives relatively good output as expected.

Table 1. Semantic words similar to man

Words	Number of trees	Measures
Woman	50	0.6236
Guy		0.5179
Men		0.5253
Person		0.5180
Lady		0.5848
Woman	200	0.6345
Guy		0.5110
Men		0.5101
Person		0.5074
Lady		0.5988
Woman	500	0.6374
Guy		0.5236
Men		0.5213
Person		0.5102
Lady		0.5960

4 Results and Discussion

In this paper, we used the dataset provided by Kaggle [15] to represent words numerically through the application of classical "Bag Of Words" model along with deep learning approaches. Different types of classifiers were used to perform the classification task. Table 2 shows the comparative structure in timing that was needed for execution of the above method using KMeans (Table 3) and Kmeans++ algorithms.

Table 2. Time taken for clustering using *K Means* and *K Means*++ algorithms

	No. of clusters	No. of trees	Time required for clustering
K Means	5	100	942 s
K Means++	5	100	613 s

Table 3. Time taken for clustering using *K Means* algorithm using different quantities of cluster and trees

No. of clusters	No. of trees	Time required for clustering
10	50	506 s
10	75	506 s
10	200	483 s
10	300	477 s
10	500	507 s

It may be noted that the KMeans++ algorithm, when applied to 10 clusters and 500 trees requires only 455.81 s for its execution. This is to be kept in mind that semantic analysis is more to be unearthed for extracting additional information. Sarcasm detection and Question detection are the in-trend areas that need much learning and concern for the betterment of the review/trend following society. The accuracy implied through these methods show very less improvement, enhancing which should be the goal of the future works.

5 Conclusion and Future Work

This paper shows the execution of a deep learning technique on Word2vec model which can further be enhanced using the Doc2vec model. Other than this, varied classifiers can be used other than the ones already mentioned above and can be judged if the optimum classifier could be found out for these type of applications. It must also be mentioned that strategic use of other clustering algorithms like DBScan, Fuzzy-C-Means, etc. can be implemented on this existing model. Last but not the least lot of experimentation can be implied in the feature selection process which can be anticipated to bring in lots of new dimensions to Deep learning approaches in Sentiment Analysis.

References

1. Bengioy, Y., Courville, A., Vincenty, P.: Representation Learning: A Review and New Perspectives, 23 April 2014. arXiv:1206.5538v3 [cs.LG]
2. Bengio, Y.: Learning deep architectures for AI. Found. TrendsR Mach. Learn. **2**(1), 1–127 (2009). https://doi.org/10.1561/2200000006
3. Liu, B.: Sentiment analysis and opinion mining. Synth. Lect. Hum. Lang. Technol. **5**, 1–167 (2012)
4. Turney, P.: Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. Paper Presented at the Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, PA (2002)
5. Somasundaran, S., Wiebe, J.: Recognizing stances in ideological on-line debates. In: Proceedings of the Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text (2010)
6. Hu, M., Liu, B.: Mining and summarizing customer reviews. Paper Presented at the Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2004)
7. Kim, S.-M., Hovy, E.: Extracting opinions, opinion holders, and topics expressed in online news media text. Paper Presented at the Proceedings of the Workshop on Sentiment and Subjectivity in Text (2006)
8. Deng, L., Wiebe, J.: Sentiment propagation via implicature constraints. In: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, pp. 377–385. EACL (2014)
9. Jo, Y., Oh, A.H.: Aspect and sentiment unification model for online review analysis. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, pp. 815–824. ACM (2011)

10. Ling, P., Geng, C., Menghou, Z., Chunya, L.: What Do Seller Manipulations of Online Product Reviews Mean to Consumers? (HKIBS Working Paper Series 070-1314) Hong Kong Institute of Business Studies, Lingnan University, Hong Kong (2014)
11. Rokach, L.: Ensemble methods for classifiers. In: Maimon, O., Rokach, L. (eds.) Data Mining and Knowledge Discovery Handbook, pp. 957–980. Springer, US (2005). http://dx.doi.org/10.1007/0-387-25465-X_45
12. Agarwal, A., Xie, B., Vovsha, I., Rambow, O., Passonneau, R.: Sentiment analysis of Twitter data. In: Proceedings of the Workshop on Languages in Social Media, pp. 30–38. Association for Computational Linguistics (2011)
13. Sharma, A., Dey, S.: A comparative study of feature selection and machine learning techniques for sentiment analysis. In: Proceedings of the 2012 ACM Research in Applied Computation Symposium, pp. 1–7. ACM (2012)
14. Kouloumpis, E., Wilson, T., Moore, J.D.: Twitter sentiment analysis: the good the bad and the omg! In: ICWSM, vol. 11, pp. 538–541 (2011)
15. Kaggle: Bag of Words Meets Bags of Popcorn. https://www.kaggle.com/c/word2vec-nlp-tutorial/data
16. Cython C-Extensions for Python. http://cython.org/

Medical Equipment Failure Rate Analysis Using Supervised Machine Learning

Rasha S. Aboul-Yazeed(✉), Ahmed El-Bialy, and Abdalla S. A. Mohamed

Systems and Biomedical Engineering Department, Faculty of Engineering,
Cairo University, Giza, Egypt
Rashasaleh24@hotmail.com

Abstract. Machine learning is widely used to identify patterns in data and to assemble models that anticipate future outcomes based on historical data. One of the critical components required for efficient healthcare services provision is medical equipment. Applying machine learning for failure rate modeling and prediction is of great importance. Therefore, two different stochastic models, ARMA and GARCH models, were utilized to analyze failure rate data. The outcome of each model was compared with previous work so that to achieve the best model that represent the failure rate data.

Keywords: Machine Learning · Time series analysis · ARMA model
GARCH model · Failure rate forecasting

1 Introduction

During two decades, Machine Learning (ML) has become one of the maintainers of information technology and with that, a rather central, but usually hidden, part of our life. With the ever increasing amounts of data becoming available, there is good reason to believe that smart data analysis will become even more pervasive as a necessary ingredient for technological progress [1].

Machine learning is used to find patterns in data and to build models that predict future outcomes based on historical data [2]. The performance and computational analysis of machine learning algorithms is a branch of statistics known as computational learning theory. Machine Learning algorithms are organized into taxonomy, based on the desired outcome of the algorithm. There are many types such as supervised, unsupervised, and reinforcement learning, etc. [3].

Supervised learning is a type of machine learning algorithm that uses a known dataset (termed the training dataset) to make predictions. The training dataset includes input data and response values. Consequently, the supervised learning algorithm seeks to build a model that can make predictions of the response values for a new dataset. A test dataset is usually utilized to validate the model. Larger training datasets are valuable, as they often yield models with higher predictive power that can be generalized well for new datasets [2].

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 319–327, 2018.
https://doi.org/10.1007/978-3-319-74690-6_32

Two categories of algorithms are included in supervised learning: **classification** for categorical response values, where the data can be separated into specific "classes" and **regression** for continuous-response values.

Supervised learning is used in: (1) Biological applications (drug discovery and tumor detection), (2) Pattern recognition applications (images and speech), (3) Energy applications (load and price forecasting) and (4) Financial applications (bond classification, algorithmic trading, and credit scoring) [2].

On the other hand, excessive medical equipment down-time due to absence of preventive maintenance, inability to repair, and lack of spare parts results in 25–35% (twenty five to thirty five percent) of equipment out of service [4].

As the core interest is patient safety, this will positively influences by continuously observing and studying the behavior of the medical equipment failure along time.

The aim of this paper is to study and complete the research in [5] that intended to reach an accurate model for failure data representation suggesting an approach for failure rate forecasting.

The paper is organized as follows: Method description is in the next section where failure data analysis, ARMA and GARCH models were introduced. Section 3 reports results and discussion. While conclusion is illustrated in Sect. 4.

2 Material and Method

Utilizing ML methods to find patterns in the failure history data of the hematology medical Equipment in [5] and to build models that represent the failure data and predict future outcomes based on historical data was our concern.

We, in this paper, present the second step or, in other words, the future work of [5] that is to apply autoregressive moving average (ARMA), and Generalized AutoRegressive Conditional Heteroskedasticity (GARCH) to the failure rate data of the laboratory medical equipment so that the outcome of each model is to be compared to the outcome of the Autoregression (AR) model who succeeded in [5] to define a highly rigorous failure forecasting model with minimum mean squared error (MMSE) less than 0.1% and succeeded to represent failure rate data of the medical equipment. The AR model could predict the failure rate occurrence every two days.

After comparing the outcome of AR, ARMA and GARCH models, we should reach the best model that represent the failure rate data in order to achieve the finest failure-forecasting model.

2.1 Data Analysis

As a brief look on the data we utilized in our research, as discussed in [5], along three years, failures history is observed and scheduled, and then the differences between two failures have been calculated. The obtained data is considered as a time series and arranged in a proper chronological order. The time series is represented here by the failure rate data. Failure rate is the time difference between two failures. Before utilizing the models, failure rate data has passed through several steps.

Data Preprocessing

Smoothing: The locally weighted scatterplot smoother (Lowess) [6] technique is used to depose any sudden variations in the failure rate data after being fitted. Such variations exist as a result of the corruption of the time series with noise, Lowess is considered as a common nonlinear correction techniques and it is more powerful compared to other nonparametric regression techniques [7].

Interpolation [8]: It is the construction of a curve $y(x)$ which passes through a set of data points (x_i, y_i) for $i = 0, 1, \ldots, n$ where the data points are $a = x_0 < x_1 < x_2 < \ldots x_{n-1} < x_n = b$. The constructed curve $y(x)$ can then be used to estimate the values of y at positions x which are between the end points a and b (interpolation) or to estimate the value of y for x exterior to the end points (extrapolation) [9]. So, it can be used to fill-in missing data, and make predictions, etc. [2].

The result from smoothing is a non-uniformly space sampled data. To reach uniform sampled data, interpolation is applied.

2.2 ARMA Model

In 1970, ARMA models became prominent by Box and Jenkins [10] and for over fifty years, the Box–Jenkins methodology using ARMA linear models have prevailed many areas of time series forecasting [11]. When modeling linear and stationary time series, one frequently chooses the class of ARMA models because of its high performance and robustness. The selection of a particular ARMA model, however, is neither easy nor without implications for the goal of the analysis [12].

Given a time series of data $\left\{x\left(t_i\right),\ i = 1, 2, \ldots, N\right\}$, ARMA model typically incorporates two portions, an autoregressive (AR) portion and a moving average (MA) portion. The model is called an ARMA(p, q) model where p is the order of the autoregressive portion and q is the order of the moving average portion.

The general expression for an ARMA-process $y(t)$ is as in Eq. (1) [13],

$$\mathrm{y}(t) = \left(\sum\nolimits_{i=1}^{p} a(i)\ .\ y(t-1)\right) + \left(\sum\nolimits_{i=0}^{q} b(i)\ .\ x(t-i)\right) + \varepsilon_t \tag{1}$$

Where:

p is the order of the AR-portion of the ARMA model;
a_1, $a_{2,\ldots,}$ a_m are the roots or coefficients of the AR-portion of the model;
q is the order of the MA-portion of the ARMA model;
b_1, $b_{2,\ldots,}$ b_n are the pools or coefficients of the MA-portion of the model;
$x(t)$ are elements of the input;
ε_t is an error.

This mathematical modeling of a time series is based on the assumption that each value of the series depends only on the sum of the previous values of the same series (according to the AR component order) and on the sum of the present and previous values of a different time series (according to the MA component order) [14]. Yet, it needs a starting value (initial input) as a trigger to start the modeling process.

There are some constraints to use ARMA model. We need what we have observed to be stable, so that we can make inferences and statements about the future. The ARMA (p, q) model is stationary provided $\left|a_p\right|$ should be less than one [15].

Unlike AR model, there are no constraints on the input data of ARMA model. Therefore, utilizing different random distributions as an input is applicable.

Selecting the structure of ARMA model

It depends on two basic steps: (1) Model Order Selection; and (2) Parameters Estimation. For model order selection, the appropriate order is usually an order just after the point at which the mean square error flattens out [16].

Parameter estimation means finding the model coefficients' values that provide the best fit to the data, or the parameters yield MMSE, considering model constraints.

Triggers: Different Random Distribution Inputs

ARMA model is initiated with four alternatives of random distribution inputs. They are Gaussian, Exponential, Poisson, and Uniform distributions. They are probability distributions that arise in a great number of real situations, and one of them may succeed to mimic the real situation yielding our failure rate pattern to reach the desired failure rate model. ARMA model order combinations are estimated to be $\{(p,q), p = 1, \ldots, 20, \mathit{and}\ q = 1, \ldots, 20\}$.

2.3 GARCH Model

GARCH models have become important tools in the analysis of time series data. These models are especially useful when the goal of the study is to analyze and forecast volatility (i.e. time-varying variance) [17].

Volatility clustering, in which large changes tend to follow large changes, and small changes tend to follow small changes, or meaning that the variance appears to be high during certain periods and low in other periods, has been well recognized in time series. This phenomenon is called conditional heteroskedasticity, and can be modeled by GARCH model proposed by Bollerslev (1986), etc. Accordingly, when a time series exhibits autoregressive conditionally heteroskedasticity, this means it has the GARCH effect [18].

The term scedastic is Greek for 'variance', which, when combined with hetero, meaning 'different', gives us heteroscedastic, or different variance. Therefore, the term 'generalized autoregressive conditional heteroskedastic' can be 'generalized autoregressive conditional different variance' [19]. It considers the variance of the current error term to be a function of the variances of the previous period's error terms, so it models the changes in variance as a function of time.

If an AR model is adhering to the error variance, the model is a GARCH model. In a standard linear regression where $y_i = \propto + \beta x_i + \epsilon_i$, when the variance of the residuals, ϵ_i is constant, we call that homoscedastic and use ordinary least squares to estimate α and β. If, on the other hand, the variance of the residuals is not constant, we call that heteroscedastic [20].

The general process for a GARCH model involves three steps: (1) Estimate a best-fitting AR model; (2) Compute variance of residuals; and (3) Test for significance.

For modeling the failure data with consideration of other external effects such as periodic preventive maintenance (ppm), the GARCH model is utilized with AR model from order 1 to 20. The residuals of failure data when applied to the model are used to test the residuals variance.

3 Results and Discussion

3.1 ARMA Model Results

ARMA model identification is considered as a challenge. Testing all combinations starting ARMA (1, 1) to ARMA (20, 20) to estimate the variation of MMSE values relative to the ARMA models order with different random distribution inputs. Figure 1 shows the different model inputs, the mean squared error for all ARMA(p, q), The system output (the real failure data) compared with the model output (ARMA model triggered with different random distribution inputs), and the final model error resulted from choosing the ARMA order having the minimum mean squared error.

Both AR and MA coefficients of different random distribution inputs are illustrated in Tables 1 and 2.

Table 1. Parameters of ARMA(p, q) with Gaussian and Uniform random distribution inputs

Gaussian		Uniform	
AR(p)	MA(q)	AR(p)	MA(q)
−2.33726598	0.00134513	−2.2772437	0.00033549
1.37424622	−0.00162043	1.32392116	−0.0002578
−0.00342074	0.00011108	−0.0017328	
0.23469965	−0.00019428	−0.0033557	
−0.26825658	0.00023414	−0.0021195	
1.0246565	0.000381567	0.06695397	
0.07493581	0.000091451	−0.106414	
0.00843225	0.000153644	−1.0048222	
−0.1549687	−0.00092473	0.0924731	
0.0190248	−0.00041736	2.164842	
		0.03819544	
		0.1275116	

ARMA model orders at which the mean square errors flatten out are ARMA (10,9), ARMA (12,1), ARMA (15,5) and ARMA (10,5) for Gaussian, Uniform, Exponential and Poisson random distribution inputs respectively.

ARMA models error ranges between different models' output and real-life data are wide. MMSE values are 0.03065168, 0.02232924, 0.01828092, and 0.0191027 for Gaussian, Uniform, Exponential and Poisson random distribution inputs respectively.

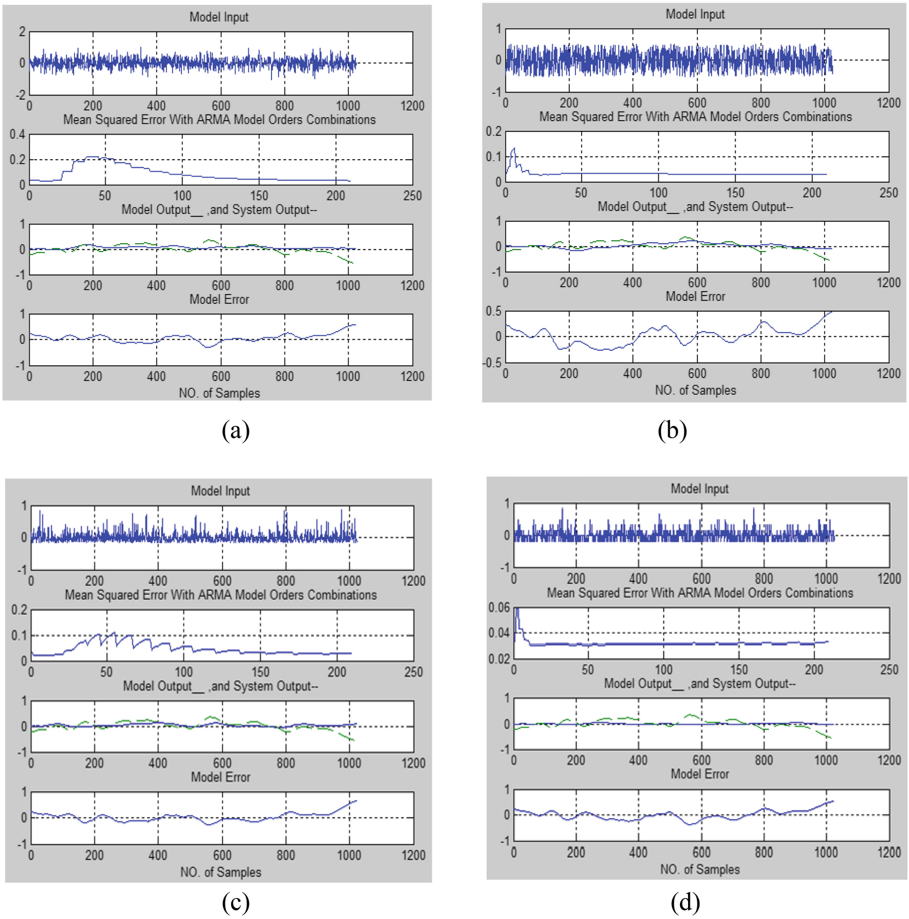


Fig. 1. ARMA model results with different random distribution inputs (a) Gaussian (b) Uniform (c) Exponential (d) Poisson

These MMSE values are greater than AR model MMSE value in [5], which was less than 0.1%.

Moreover, MA coefficients of the tested ARMA models are very small. This means that Gaussian, Uniform, Exponential, and Poisson input ARMA models are approaching to be AR model.

3.2 ARCH Model Results: Testing Residuals Variance

Whenever a time series is said to have GARCH effects, the series is heteroskedastic, i.e., its variances vary with time. If its variances remain constant with time, the series is homoskedastic [21].

Table 2. Parameters of ARMA(p, q) with Exponential and Poisson random distribution inputs

Exponential		Poisson	
AR(p)	MA(q)	AR(p)	MA(q)
−2.3358879	−0.0016328	−2.338527	0.00004338
1.37049889	0.0018877	1.37660488	0.0006405
−0.0004075	0.00045814	−0.0031911	−0.0004621
0.234585	−0.0002529	0.23165895	−0.0001513
−0.268785	−0.0002687	−0.2665414	−9.46E-06
−1.9246812	−8.097E-05	−1.9282457	−5.888E-05
0.01279554		−0.064875	
2.0039246		0.5246151	
0.5348169		0.06825214	
−0.0318913		0.00931575	
−2.0618955			
1.0129547			
0.38446952			
0.93174526			
0.36917852			

Therefore, the residuals of failure data have been calculated to test the variance of residuals for all model orders from 1 to 20. The residuals variance results are shown in Fig. 2. Results have shown the absence of variance in the residuals values as it ranges between 0.0018313629 to 0.0029096447. This violates the heteroscedasticity condition or the different variance condition and the series is homoskedastic.

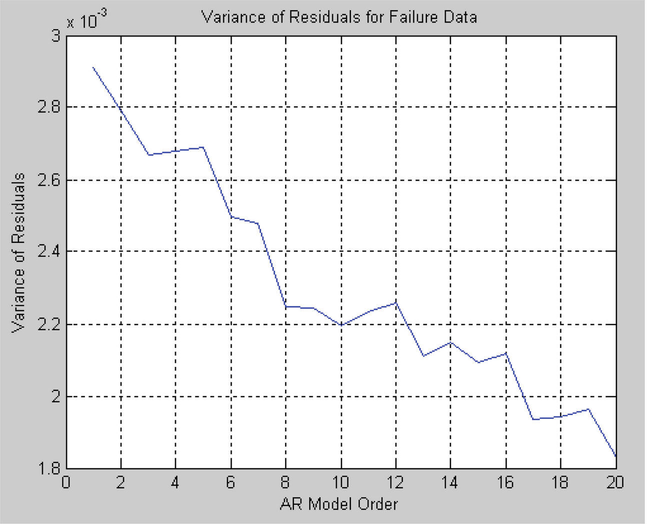


Fig. 2. Residuals variance with model orders for failure data.

4 Conclusion

Machine Learning is a specialized sub-field of Artificial Intelligence where algorithms can learn and improve themselves by studying high volumes of available data to create better computing models [22].

Utilizing AR model in [5], researchers could define a highly rigorous failure-forecasting model with MMSE less than 0.1%. The model could predict the failure rate occurrence for some medical laboratory equipment every two days.

For continuing this research, time series analysis using two different stochastic models, ARMA and GARCH, were applied to the failure rate data, and the outcome of each model were compared with [5] so that to achieve the best model that will represent the failure rate data.

Although ARMA model is supposed to be more close to the real situations of failures occurrence than AR model because of the absence of constraints on the input data, unfortunately, it could not analyze the failure rate and failed to represent a model. This is due to several reasons: (1) MMSE of ARMA model is greater than in AR model; (2) MA coefficients are very small, which means that ARMA model is approaching to be AR model; and (3) One of the benefits supposed to be in ARMA model is lower order than AR model, which does not occur. Consequently, it affects time consumption.

GARCH is a time series modeling technique that uses past variances and past variance forecasts to forecast future variances [23].

Failure data residuals variance results have shown the absence of variance in the residuals. This violates the heteroscedasticity condition and the series is homoskedastic. Therefore, the GARCH model is not applicable to model the failure data.

References

1. Alex, S., Vishwanathan, S.V.N.: Introduction to Machine Learning, 1st edn. Cambridge University Press, Cambridge (2008)
2. MathWorks, Signal Processing Toolbox: Documentation (R2016a). Retrieved September 2017
3. Yagang, Z.: New Advances in Machine Learning. InTech, London (2010)
4. Andreas, L., Caroline, T., Willi, K., Manjit, K.: How to Organize the Maintenance of Your Healthcare Technology, pp. 14–30. Ziken International (2003)
5. Rasha, S.A., Ahmed, E., Abdalla, S.A.M.: Prediction of medical equipment failure rate: a case study. In: Proceedings of the International Conference on Advanced Intelligent Systems and Informatics. Advances in Intelligent Systems and Computing, pp. 650–659. Springer, Cham (2017)
6. Box, G.E.P., Jenkins, G.M., Reinsel, G.C.: Time Series Analysis: Forecasting and Control. Wiley, San Francisco (2008)
7. John, A.B., Sampsa, H., Anna-Kaarina, J., Henrik, E., Sanjit, K.M., Jaakko, A.: Optimized LOWESS normalization parameter selection for DNA microarray data. BMC Bioinf. **5**, 194 (2004)
8. Robert, B.N.: Introduction to Instrumentation and Measurements, 2nd edn. CRC Press, Taylor & Francis, New York (2005)

9. Heinbockel, J.H.: Numerical Methods for Scientific Computing. Trafford Publishing, Victoria (2005)
10. Box, G.E.P., Jenkins, G.M.: Time Series Analysis: Forecasting and Control. Holden-Day, San Francisco (1970)
11. Rojasa, I., Valenzuelab, O., Rojasa, F., Guillena, A., Herreraa, L.J., Pomaresa, H., Marquezb, L., Pasadasb, M.: Soft-computing techniques and ARMA model for time series prediction. Neurocomputing **71**, 519–537 (2008)
12. Box, G.E.P., Jenkins, G.M., Reinsel, G.C.: Time Series Analysis: Forecasting and Control. Prentice-Hall, Englewood Cliffs (1994)
13. Atsalakis, S.G., Parasyri, G.M., Zopounidis, D.C.: Milk production forecasting by a neuro-fuzzy model. In: Research Topics in Agricultural and Applied Economics, vol. 3, pp. 3–11. Bentham Science Publisher (2012)
14. de Smith, M.S.: STATSREF: Statistical Analysis Handbook. A Web-Based Statistics Resource. The Winchelsea Press, Winchelsea (2015)
15. Anna, M.: Course materials for 14.384 Time Series Analysis, Fall 2007. MIT OpenCourseWare (http://ocw.mit.edu), Massachusetts Institute of Technology. Downloaded on 24 September 2017
16. Ramdane-Cherif, Z., Naït-Ali, A., Motsch, J.F., Krebs, M.O.: An autoregressive (AR) model applied to eye tremor movement, clinical application in Schizophrenia. J. Med. Syst. **28**, 489–495 (2004)
17. Robert, E.: GARCH 101: an introduction to the use of ARCH/GARCH models in applied econometrics. J. Econ. Perspect. **15**, 157–168 (2001)
18. Wang, W., Van Gelder, P.H.A.J.M., Vrijling, J.K., Ma, J.: Testing and modeling autoregressive conditional heteroskedasticity of streamflow processes. Nonlinear Process. Geophys. **12**, 55–66 (2005). European Geosciences Union
19. Raihan, S.: Assessing the Implications from Trade Liberalisation: Use of Different Methods and their Limitations. Economic Affairs Division of the Commonwealth Secretarial, London (2004)
20. Rob, R.: Volatility Forecasting I: GARCH Models. New York Courant Institute of Mathematical Sciences, New York University (2014)
21. David, E., Joakim, S.: Evaluating VaR with the ARCH/GARCH Family. Bachelor thesis, Uppsala University (2011)
22. Parmita, G.: The Value of Machine Learning: Benefits and Best Practices. Dataversity Education, LLC (2017)
23. Martins, Y.O., Isiguzo, E.A., Abubakar, S.M., John, J.M.: Conditional heteroscedasticity in streamflow process: paradox or reality? J. Mod. Hydrol. **2**, 79–90 (2012)

Big Data and Classification

Big-Data Aggregating, Linking, Integrating and Representing Using Semantic Web Technologies

Abeer Saber[1,2(✉)], Aya M. Al-Zoghby[2], and Samir Elmougy[2]

[1] Department of Computer Science, Faculty of Computers and Information,
Kafr El-Sheikh University, Kafr El-Sheikh 33511, Egypt
Abeer_Saber@fci.kfs.edu.eg

[2] Department of Computer Science, Faculty of Computers and Information,
Mansoura University, Mansoura 35516, Egypt
{aya_el_zoghby,mougy}@mans.edu.eg

Abstract. Semantic web provides information for humans as well as computers to semantically maintain a large-scale of data and provide a meaningful content of unstructured data. It offers new benefits for big-data research and applications. Big data is a new term refers to a massive collection of datasets from various sources in structured, semi-structured, and unstructured data collection. Their integration faces many problems such as the structural and the semantic heterogeneity as the processing of these data is difficult using traditional databases and software techniques. In this paper, the data resources are extracted and aggregated from different sources on the web following by using the geospatial ontology to transform this data into RDF format. RDF format is used to integrate the data semantically and construct the big-data semantic model that is used to store data. The major contribution of this research is to aggregate, integrate, and represent geospatial data semantically. A case study of cities data is used to illustrate the proposed workflow functionalities. The main result of this research is to solve the heterogeneous problem in different data sources with improving the data aggregation, integration, and representation.

Keywords: Semantic web · Big-data · Ontology
Extract-Transform-Load process · Geospatial data · Semantic heterogeneity
Structural heterogeneity

1 Introduction

Data, as text, audio, video, and images, are published on the web using HyperText Markup Language (HTML). HTML is poor in defining and formalizing the meaning of its context. Semantic Web (SW) is a mesh of data that represents the meanings through connectivity, expressing multiple viewpoints, and using logical rules to share information across applications [1, 2]. It offers information to humans and computers to manipulate large-scale of data semantically. Therefore, SW beats the problems of formatting data in a suitable format to take advantage in information retrieval process.

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 331–342, 2018.
https://doi.org/10.1007/978-3-319-74690-6_33

Ontologies are a base block of building SW technologies in which its metadata schemas provide a controlled vocabulary of concepts [3]. The ontology describes data semantics that represents the background knowledge on the semantic level [4]. The semantic level is a collection of semantic entities including its concepts and relations instead of simple words that are applied in the thesaurus. It specifies the relations among entities and holds the facts and rules about the scope of the problem. The usage of ontology is vital to integrate thinking[1] and intelligent big data retrieval and hence introduce new thoughts in its related fields and applications.

Big data refers to a large scale that is used to represent a huge collection of datasets which are integrated from several sources in a structured, semi-structured, and unstructured data collection. Although unstructured and semi-structured data are at least "85%" of the whole data, both types are difficult to be processed using the traditional tools because most of these data are inaccessible to users [5, 6]. Big-data faces many technical challenges such as (1) Acquire and record data, (2) extract and clean information, (3) integrate, aggregate, and represent data, (4) query processing and analysis, (5) interpretation, and (6) privacy and security [7].

While describing, integrating, and interpreting heterogeneous data can be robust in big-data context, its handling faced challenges due to the variety of data formats and the kind of the existed knowledge [8].

Big data integration means large linkage volumes of heterogeneous data from different sources [9]. Also, data sources are dynamic and heterogeneous in their structure. Schema mapping, record linkage, and data fusion are the main challenges facing it [10].

In this paper, we propose and implement a new workflow model to enhance the data aggregation, integration and representation using semantic web technologies with overcoming semantic heterogeneities problem that is appeared because of using various expressions for the same things or when using the same terms for different things. Section 2 presents some related work. Section 3 introduces a case study and a motivating example for geospatial data integration. Section 4 presents the proposed workflow and its implementation. Experimental analysis and conclusions are discussed in Sects. 5 and 6 respectively.

2 Related Work

In big data, the problem of structural and semantic heterogeneity revolted a great concern to researchers all over the world as it causes many problems in data extraction, aggregation, and integration. Organizing data are necessary to make big data query engines and analytic tools more efficient and creative. So, the concepts that have the same meaning must be connected through links, and distinct concepts must be represented via semantic metadata [11].

[1] For e.g., a serious method to conceptualize field knowledge and modeling that can be used to describe the data semantics.

2.1 Semantic Web for Improving ETL Process

Many research papers were depending on SW and metadata to solve data integration and representation problems either in Extract-Transform-Load (ETL) process that is considered one of the prevalent approaches to data integration or through proposing a new technique or tool. Bergamaschi et al. [12] proposed a tool that enhanced the definitions of ETL by allowing semi-automatic inter-attribute semantic transforming through recognizing schemas of data sources then grouping attribute values semantically. Jiang et al. [13] integrated heterogeneous data from database sources semantically after mapping it to data warehouse according to the domain ontology. Their model assumed that the type of data sources is relational database only. Huang [14] proposed an ontology-based ETL method for the structured data sources to automatically extract marine data from different formats following by transforming their various schemes into unified schemes according to the integrated database. However, this method did not solve the unstructured data problems. Bansal and Kagemann [11] and Bansal [6] integrated and published data from structured data sources as linked open data through adding semantic data model and semantic data instance to the transformation layer using Web Ontology Language (OWL), Resource Description Framework (RDF), and Simple Protocol and RDF Query Language (SPARQL) technologies.

2.2 Semantic Web for Data Integration, Aggregation, and Representation

As for the geospatial data integration domain, Cruz et al. [15] extracted geographic data from web tables by extracting and identifying rich tables features and then constructing a schema and instances using RDF. In their work, the table extraction depends on the <table> tag, and some tables use <div> tag while some other tables are not formatted properly. Zhang et al. [16] presented a semantic approach that extracts, links, and integrates structured geospatial data from heterogeneous sources. It first extracts data to transforms it into RDF and then links and integrates it using linking and integration algorithms. However, the transformation of the unstructured data file into RDF file is still rare in research. Boury-Brisset [17] designed a global architecture for intelligence data integration. Its main components are ingestion process, ontology support, semantic enrichment, and interactions with other reasoning modules with transforming different data, which are acquired from various sources, into useful, actionable intelligence promptly. However, this work is still immature, and the experimentations showed that incremental development and testing stages performance are still required to be improved.

Accessing domain resources on the web or the mobile networks is difficult because they are heterogeneous, decentralized. Xiong et al. in [18] used an ontology to solve this problem and integrate educational resources successfully. For big data aggregation problem, Gollapudi [19] built an architecture that combined the data lake with semantic computation concepts. Data can be aggregated from various resources such that no assumption is provided related to knowing in the way, place, or time that data could be used. Kang et al. [5] constructed a big data semantic model in line with MapReduce framework to store data semantically and to overcome the problem of understanding between heterogeneous data systems. This model did not have the process of data integration from existing database system.

In 2010, Saradha [20] tried to overcome the problem of unstructured web data format but the used search engines return inappropriate results, and the information available on the web was difficult to be integrated. Also, search engines take care of placements rather than information semantics that used converting conventional tourism data into RDF format. In other words, it is not an appropriate solution as it depends on downloading and converting HTML to XML followed by converting XML to RDF. Jadhao et al. [21] presented a module to enables extracting information from unstructured data and presenting it using innovative graph mechanism through two steps. First, information is extracted to identify entities and relations from unstructured text. Second, the extracted information is represented in RDF format and then visualizes the query results using the spring graph technique.

Semantic data integration from different data sources research is still rare and faces many problems such as semantic heterogeneities Here; we aim to overcome this problem and to improve the data aggregation, integration, and representation using semantic web technologies. Next section presents a case study of semantic heterogeneity between various data sources.

3 A Case Study

To illustrate the proposed workflow functionality, a case study of cities data is used. Data resources are aggregated from different geospatial data resources on the internet such as *MapCruzin group* [22], *Data.gov* [23], *United States Census* [24], *OST/SEC Map group* [25], *USCitiesList.org* [26], and *Gaslamp media* [27]. Table 1 represents the semantic heterogeneity in these sources. Some data in these resources are the same but are referred to use different names such as (city, name), (Aland, land area), (Awater, water area), (country_fip, countryFP, countryfips), (LON, longitude), and (LAT, latitude).

Table 1. Semantic heterogeneity in different data sources.

Attribute Name / *Data Resource*	Country	City	Longitude	*Latitude*	*State*	*Water Area*
Gaslamp media	Country	City	Longitude	Latitude	State	…….
Data.gov	Country	Name	Longitude	Latitude	State	……..
OST/SEC group	……......	Name	LON	LAT	ST	……..
US Census	CountryFP	Name	………	……	StateFP	Awater
USCities org	Country	Name	Longitude	Latitude	state	Water _area

4 The Proposed Workflow

4.1 Methodology

The proposed workflow, shown in Fig. 3, aims to aggregate different geospatial data resources from the web semantically and to integrate the extracted resources data semantically to store it as a semantic big data geospatial model. The proposed workflow consists of four main components. The first component is *Data resources aggregation* in which it takes the geospatial ontology and metadata as input, as shown in Fig. 1, to aggregate geospatial data resources from different resources over the internet. According to the format of the data aggregated from these various resources, the second or third component will be applied to transform it into an RDF file.

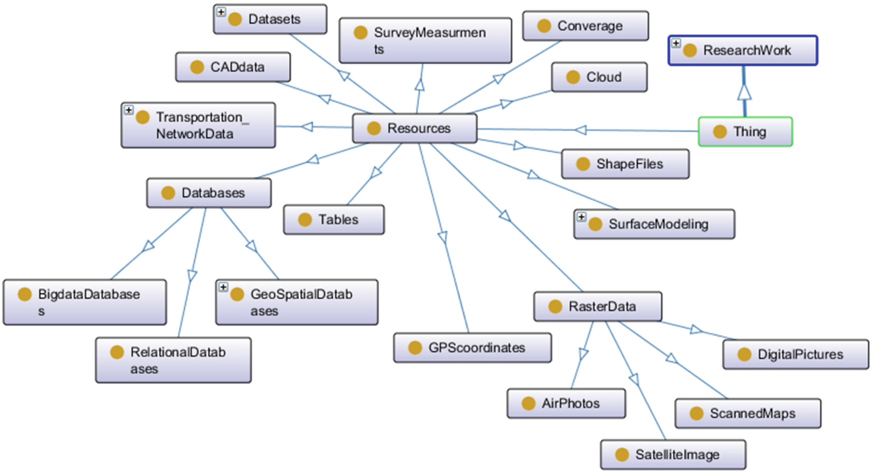


Fig. 1. Geospatial ontology used for heterogeneous resources aggregation

In the case of the structured or semi-structured data resource, the second component (RDF generation of structured and semi-structured data resources) will be applied. In this component, the data is extracted and transformed into RDF files depending on the generic geospatial ontology shown in Fig. 5. RDF generation algorithm is used for this transformation as shown in Fig. 4. This algorithm is created using the alignment API[2] To transform CSV data file into RDF according to the applied generic geospatial ontology. Thereby, the structured and semi-structured data files like (XML, EXCEL, JSON, and so forth) are transformed into CSV data file before the implementation of the RDF generation algorithm. In the case of the unstructured data resource, the third component (RDF generation for unstructured data resources) adopted from [21] is applied. In this component, structuring analysis is used to remove the noisy elements

[2] The alignment API 4.0 [28, 29].

```
Input: RDF file: $source_1$, RDF file: $source_2$
Output: Same_attributes[] MatchedAttributes
//Stage1:
Repository $\leftarrow$ $source_1$;
Repository $\leftarrow$ $source_2$;
Attributes[] $attr\_s_1$ $\leftarrow$ extract joint attributes from $source_1$;
Attributes[] $attr\_s_2$ $\leftarrow$ extract joint attributes from $source_2$; // Stage2:
For all attributes in $attr\_s_1$
{
    For all attributes in $attr\_s_1$
    {
          If   $\exists$(an attribute $\in$ $attr\_s_1$(A))=the same name with(an attribute $\in$ $attr\_s_2$ (B))
                 Then Add B in MatchedAttributes;
          Else If there is an attribute $\in$ $attr\_s_1$ (A) has the same alias name with attribute $\in$ $attr\_s_2$ (B)
                 Then Add B in MatchedAttributes;
    }
}// Stage3:
Display MatchedAttributes from $source_2$;
```

Fig. 2. Data linking algorithm

and generate metadata information followed by an information extraction process that has two processes: linguistic analysis and semantic analysis.

The linguistic analysis process has two phases. The first phase includes sentence splitting, part of speech tagger, morphological analyzer, JAPE transducer, and onto root gazetteer. JAPE transducer executes specified rules over the annotated corpus based on regular expression. Onto root gazetteer is responsible for taking the domain ontology as an input to produce an annotated corpus with the geospatial entities. The second process is a semantic analysis, which is applied to capture the unknown relations appeared in the textual data between the annotated entities. The linguistic analysis output is used as the input of the semantic analysis process which uses simple semantic rules to extract the relations from unstructured textual data.

The last component is the semantic model for geospatial data. In this component, data linking algorithm presented in Fig. 2, is used to link RDF data files semantically before integrating to overcome the semantic heterogeneity problem. Next, applying linking and integration algorithm to compare and prevent redundancy [16], followed by integrating all RDF files data into one file. Finally, the big data semantic model is constructed to store the data semantically.

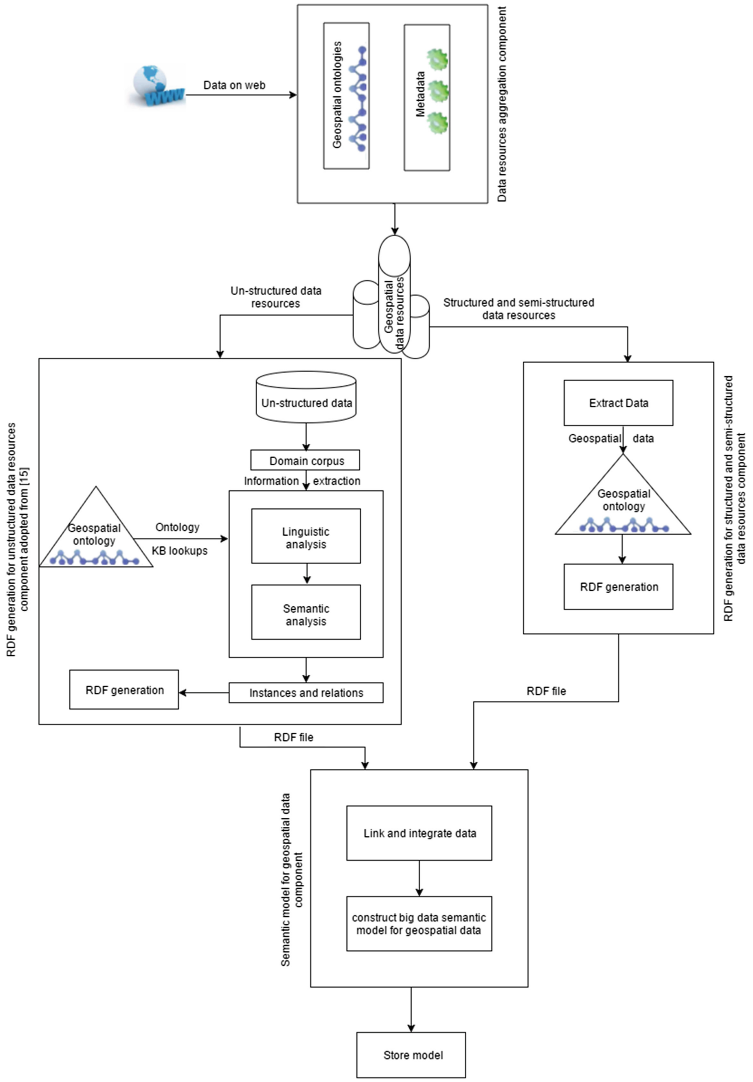

Data on web
Geospatial ontologies
Metadata
Data resources aggregation component
Geospatial data resources
Un-structured data resources
Structured and semi-structured data resources
RDF generation for unstructured data resources component adopted from [15]
Un-structured data
Domain corpus
Information extraction
Geospatial ontology
Ontology
KB lookups
Linguistic analysis
Semantic analysis
Instances and relations
RDF generation
Extract Data
Geospatial data
Geospatial ontology
RDF generation
RDF generation for structured and semi-structured data resources component
RDF file
RDF file
Semantic model for geospatial data component
Link and integrate data
construct big data semantic model for geospatial data
Store model

Fig. 3. The proposed workflow

5 Experimental Analysis

5.1 The Experimental Setup

Some of SW technologies are used to implement our proposed workflow as follows:

1. Uniform Resource Identifier (URI): It is the standard to identify and locate resources such as web pages, providing a baseline to represent the characters used in most of the world’s languages and identifying resources [30].

```
Input: CSV file: $source_1$, Ontology file: $source_2$
  Output: 1- RDF data file generation
          2- Measuring similarities between entities in
             $source_1$ and $source_2$

  //Step1: Write CSV data as RDF format

  File CSVFile ← read $source_1$ data

RDF class name= $source_1$ name

  While(CSVFile)
  {
      If(linecount == 1)
      {For all column name in CSVFile
            {DataProperty ← column name}
      }
      Else
      {For each line in CSVFile
            {Set it as a new individual}
      }

  }//step2

  OntModel geospatial_ontology ← read $source_2$

  //Step3: Using Alignment API to measure similarities

  Source ontology <- the ontology created by $source_1$

  Target ontology <- $source_2$

  // Setting up the alignment process

  AlignmentProcess.init(SourceOntology, TargetOntology);

  AlignmentProcess.align(null, new Properties());

  Print $source_1$ as RDF after alignment process
```

Fig. 4. RDF generation algorithm

2. RDF: Data exchange model on the internet, which describes website metadata and delivers interoperability among applications. It facilitates data merging of different schemas and permits structured and semi-structured data to be mixed, exposed, and shared across various applications [31, 32].
3. OWL: It is an SW language built on the top of RDF. It is written in XML that represents things, groups of things, and relations between things knowledge.
4. SPARQL [34]: It is a query language and protocol for RDF used to query, retrieve, and process data in RDF format [33].
5. Alignment API: It provides abstractions for network notations of ontologies, alignments, and correspondences, as well as building blocks for manipulation such as matches, evaluators, renderers, and parsers [28].
6. Eclipse program.

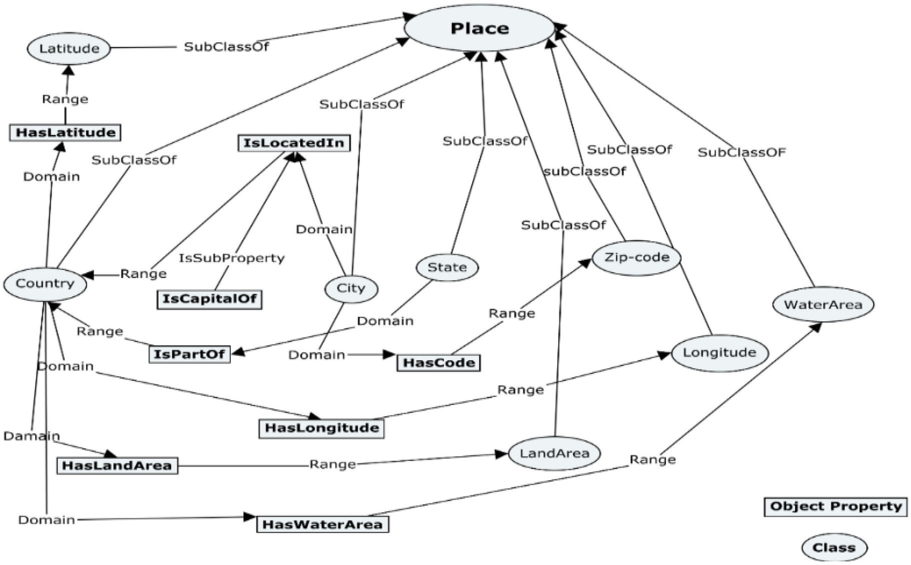
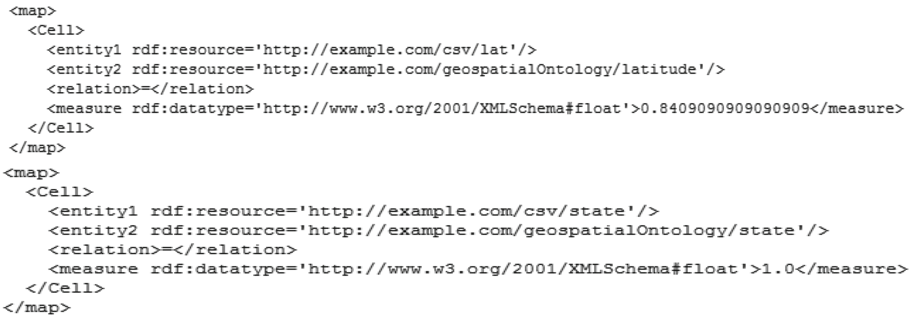


Fig. 5. Generic geospatial ontology

7. Protégé "ontology editor": It is an open source editor to build domain models and knowledge-based applications with ontologies [35].

5.2 Experiments and Results

We have solved the semantic heterogeneity problem that appeared in data integration using the proposed RDF generation algorithm that depends on the *Alignment API* and generic geospatial ontology to catch the similarities between the ontology and RDF data properties as shown in Fig. 6[3].

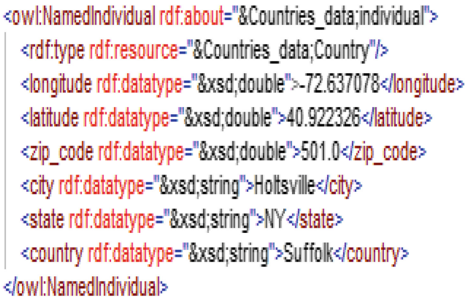


```
<map>
  <Cell>
    <entity1 rdf:resource='http://example.com/csv/lat'/>
    <entity2 rdf:resource='http://example.com/geospatialOntology/latitude'/>
    <relation>=</relation>
    <measure rdf:datatype='http://www.w3.org/2001/XMLSchema#float'>0.8409090909090909</measure>
  </Cell>
</map>
<map>
  <Cell>
    <entity1 rdf:resource='http://example.com/csv/state'/>
    <entity2 rdf:resource='http://example.com/geospatialOntology/state'/>
    <relation>=</relation>
    <measure rdf:datatype='http://www.w3.org/2001/XMLSchema#float'>1.0</measure>
  </Cell>
</map>
```

Fig. 6. Matching attributes between data and ontology

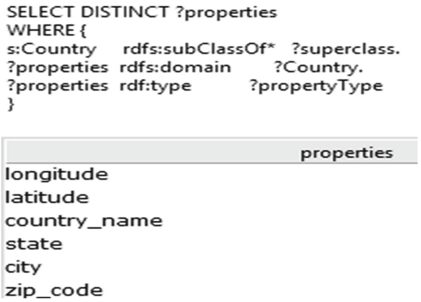
[3] The data used in Fig. 6 are extracted from [25].

We used CSV data presented in [27] and converted it into RDF data file using our proposed RDF generation algorithm. An example of the generated RDF data file is shown in Fig. 7. If the data is in unstructured, the attributes from its RDF file are extracted using SPARQL query then the proposed *Data Linking Algorithm* is applied to match the entities. To test the algorithm, the data presented in [27] and [22] are converted to RDF format; Fig. 8 shows an example to clarify this process. We repeated the same operations on the data given in [22] to extract its attributes.



```
<owl:NamedIndividual rdf:about="&Countries_data;individual">
  <rdf:type rdf:resource="&Countries_data;Country"/>
  <longitude rdf:datatype="&xsd;double">-72.637078</longitude>
  <latitude rdf:datatype="&xsd;double">40.922326</latitude>
  <zip_code rdf:datatype="&xsd;double">501.0</zip_code>
  <city rdf:datatype="&xsd;string">Holtsville</city>
  <state rdf:datatype="&xsd;string">NY</state>
  <country rdf:datatype="&xsd;string">Suffolk</country>
</owl:NamedIndividual>
```

Fig. 7. Example of RDF converted file

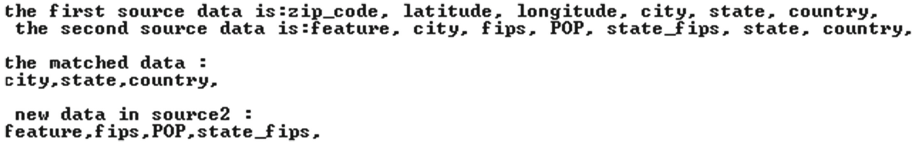


```
SELECT DISTINCT ?properties
WHERE {
s:Country      rdfs:subClassOf*   ?superclass.
?properties   rdfs:domain        ?Country.
?properties   rdf:type           ?propertyType
}
```

properties
longitude
latitude
country_name
state
city
zip_code

Fig. 8. Attributes extraction using Sparql

The next step is to match the extracted attributes from both of [27] and [22] as shown in Fig. 9. Since *name* and *city* attributes are referring to the same data, they have matched attributes. This means that the semantic heterogeneity problem is solved and the data resources are integrated and stored semantically as presented in [5].



```
the first source data is:zip_code, latitude, longitude, city, state, country,
 the second source data is:feature, city, fips, POP, state_fips, state, country,

the matched data :
city,state,country,

 new data in source2 :
feature,fips,POP,state_fips,
```

Fig. 9. Example of matched attributes

5.3 Evaluation

Working in Big data study is usually focusing on Volume, Velocity, and Variety areas. Semantic web technology was not born for big-data. Velocity and volume are still posing a big challenge for SW technologies. Our model solves the semantic heterogeneity problem in data integration caused by the variety of big data. The output of the system is indicating which elements of the input schemas are matched for integrating them successfully.

The trust of the alignment provider in the relationship between attributes, the greater the value, the higher the confidence (Measurement value: float between 0.0 and 1.0). Different matching schemas used in the alignment API are discussed in [36].

6 Conclusions

This work discussed a new hybrid workflow that permits the user to be aggregate, link, integrate, and represent the geospatial data from various sources using semantic technologies. First, the geospatial data resources are aggregated semantically and then integrated semantically using a generic geospatial ontology to solve the semantic and the structural heterogeneity problems that hinder the data integration process. Finally, data are represented and stored semantically. Semantic web technology solves the variety problem in big data, but it can’t solve the problem of volume. For the semantic heterogeneity problem of connecting entities between data sources, we are working to use some common attributes to improve the manipulating of data linking and improve the integration results.

References

1. Wu, H., Yamaguchi, A.: Semantic web technologies for the big data in life sciences. Biosci. Trends **8**(4), 192–201 (2014)
2. Ahmed, Z., Gerhard, D.: Web to Semantic Web & Role of Ontology (2010). arXiv preprint: arXiv:1008.1331
3. Jain, V., Singh, M.: Ontology-based information retrieval in semantic web: a survey. Int. J. Inf. Technol. Comput. Sci. (IJITCS) **5**(10), 62 (2013)
4. Di Martino, B., Esposito, A., Nacchia, S., Maisto, S.A.: A semantic model for business process patterns to support cloud deployment. Comput. Sci. Res. Dev. **32**(3–4), 257–267 (2017)
5. Kang, L., Yi, L., Dong, L.: Research on construction methods of big data semantic model. In: Proceedings of the World Congress on Engineering (WCE 2014), vol. 1, London, UK (2014)
6. Bansal, S.K.: Towards a semantic extract-transform-load (ETL) framework for big data integration. In: 2014 IEEE International Congress on Big Data (BigData Congress), Anchorage, pp. 522–529. IEEE (2014)
7. Bertino, E.: Big data – opportunities and challenges. In: IEEE 37th Annual Computer Software and Applications Conference, Kyoto, Japan, pp. 479–480 (2013)
8. Thirunarayan, K., Sheth, A.: Semantics-empowered approaches to big data processing for physical-cyber-social applications. In: Semantics for Big Data: Papers from the AAAI Symposium. AAAI Technical report FS-13-04, Arlington, Virginia, USA, pp. 68–75 (2013)
9. Arputhamary, B., Arockiam, L.: A review on big data integration. Int. J. Comput. Appl., 21–26 (2014)
10. Bizer, C., Boncz, P., Brodie, M.L., Erling, O.: The meaningful use of big data: four perspectives–four challenges. ACM SIGMOD Rec. **40**(4), 56–60 (2012)
11. Bansal, S.K., Kagemann, S.: Integrating big data: a semantic extract-transform-load framework. Computer **48**(3), 42–50 (2014)
12. Bergamaschi, S., Guerra, F., Orsini, M., Sartori, C., Vincini, M.: A semantic approach to ETL technologies. Data Knowl. Eng. **70**(8), 717–731 (2011)
13. Jiang, L., Cai, H., Xu, B.: A domain ontology approach in the ETL process of data warehousing. In: 2010 IEEE 7th International Conference on e-Business Engineering (ICEBE), Shanghai, pp. 30–35 (2010)

14. Huang, O.R., Du, Y.L., Zhang, M.H., Zhang, C.: Application of ontology-based automatic ETL in marine data integration. In: IEEE Symposium on Electrical & Electronics Engineering (EEESYM), Kuala Lumpur, Malaysia, pp. 11–13 (2012)
15. Cruz, I.F., Ganesh, V.R., Mirrezaei, S.I.: Semantic extraction of geographic data from web tables for big data integration. In: Proceedings of the 7th Workshop on Geographic Information Retrieval, Orlando, FL, USA, pp. 19–26. ACM (2013)
16. Zhang, Y., Chiang, Y.Y., Szekely, P., Knoblock, C.A.: A semantic approach to retrieving, linking, and integrating heterogeneous geospatial data. In: Joint Proceedings of the Workshop on AI Problems and Approaches for Intelligent Environments and Workshop on Semantic Cities, pp. 31–37. ACM (2013)
17. Boury-Brisset, A.-C.: Managing semantic big data for intelligence. In: STIDS, pp. 41–47 (2013)
18. Xiong, J., Liu, Y., Liu, W.: Ontology-based integration and sharing of big data educational resources. In: IEEE 11th Web Information System and Application Conference (WISA), Tianjin, China, pp. 245–248 (2014)
19. Gollapudi, S.: Aggregating financial services data without assumptions: a semantic data reference architecture. In: 2015 IEEE International Conference on Semantic Computing (ICSC), Anaheim, CA, USA, pp. 312–315 (2015)
20. Saradha, A.: Semantic integration of heterogeneous web data for tourism domain using ontology-based resource description language. J. Comput. Appl. **3**(3), 1 (2010)
21. Jadhao, H., Aghav, D.J., Vegiraju, A.: Semantic tool for analysing unstructured data. Int. J. Sci. Eng. Res. **3**(8), 1–7 (2012)
22. MapCruzin data Homepage. http://www.mapcruzin.com/. Accessed 21 Oct 2017
23. DATA.GOV Homepage. https://catalog.data.gov/. Accessed 20 Oct 2017
24. United States Census Homepage. https://www.census.gov/. Accessed 1 Oct 2017
25. OST/SEC Homepage. http://www.nws.noaa.gov/. Accessed 20 Oct 2017
26. Cities data Homepage. https://www.uscitieslist.org/. Accessed 19 Oct 2017
27. Gaslamp media Homepage. https://www.gaslampmedia.com. Accessed 19 Oct 2017
28. David, J., Euzenat, J., Scharffe, F., Trojahn dos Santos, C.: The alignment API 4.0. Semant. Web Interoperability Usability Appl. **2**(1), 3–10 (2011)
29. Euzenat, J.: An API for ontology alignment. In: International Semantic Web Conference, pp. 698–712. Springer, Heidelberg (2004)
30. Matthews, B.: Semantic web technologies. E-learning **6**(6), 8 (2005)
31. RDF Homepage. https://www.w3.org/RDF/. Accessed 15 Oct 2017
32. RDF Homepage. http://www.webopedia.com/TERM/R/RDF.html. Accessed 1 Oct 2017
33. OWL Homepage. https://www.w3.org/2001/sw/wiki/OWL. Accessed 21 Oct 2017
34. SPARQL Query Language for RDF Homepage. https://www.w3.org/TR/rdf-sparql-query/. Accessed 20 Oct 2017
35. Protégé Homepage. http://protegewiki.stanford.edu/wiki/Main_Page. Accessed 19 Oct 2017
36. Do, H.H., Melnik, S., Rahm, E.: Comparison of schema matching evaluations. In: Net. ObjectDays: International Conference on Object-Oriented and Internet-Based Technologies, Concepts, and Applications for a Networked World, pp. 221–237. Springer, Heidelberg (2002)

Computer Aided Diagnostic System for Automatic Detection of Brain Tumor Through MRI Using Clustering Based Segmentation Technique and SVM Classifier

Atanu K. Samanta and Asim Ali Khan(✉)

EIE Department, SLIET Longowal, Punjab, India
asimsliet@gmail.com

Abstract. Due to the acquisition of huge amount of brain tumor magnetic resonance images (MRI) in the clinics, it is very difficult for the radiologists to manually interpret and segment these images within a reasonable span of time. Computer-aided diagnosis (CAD) systems increase the diagnostic abilities of radiologists and reduce the elapsed time for perfect diagnosis. An intelligent computer-aided technique is proposed in this paper for automatic detection of brain tumor from MR images. The proposed technique uses following computational methods; the K-means clustering for segmentation of brain tumor from other brain parts, extraction of features from this segmented brain tumor portion using gray level co-occurrence Matrices (GLCM), and the support vector machine (SVM) to classify input MRI images into normal and abnormal. The whole work is carried out on 64 images consisting of 22 normal and 42 images having brain tumor (benign and malignant). The overall classification accuracy using this method is found to be 99.28% which is significantly good.

Keywords: Brain tumor · Computer-aided diagnostic (CAD) system
Gray-level co-occurrence matrix (GLCM) · K-means clustering
Support vector machine (SVM) · Tumor segmentation

1 Introduction

A brain tumor is an abnormal mass of tissue associated with brain or central nervous system that can hamper proper brain activities. The masses grow rapidly in an uncontrolled way. A primary brain tumor originates from brain cells. Secondary brain tumors are originated from other human organs which have been metastasized and thus affect the normal function of brain [1]. Central nervous system tumor contributes a distinct type of diseases that vary from benign, slow-growing lesion to malignant which are more aggressive in nature and can lead to death. Each of these tumors has unique pathological, radiological, and anatomic characteristics. Benign tumors grow gradually and do not spread widely. Malignant tumors grow more quickly compared to benign tumors and can spread to surrounding tissues. Once the radiological practitioner roughly suspects brain tumor, a thorough radiological evaluation procedure has to be performed to locate the position, the area of enclosure of the affected portion and its connectedness to the surrounding tissue structures [2]. Magnetic resonance imaging (MRI) is the most

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 343–351, 2018.
https://doi.org/10.1007/978-3-319-74690-6_34

appreciated and popular brain imaging technique for proper diagnosis of brain tumor patients. Its multiplane capacity, higher contrast resolution and various user friendly conventions help to diagnose the tumor location and extent [2]. Different studies on computer-aided technologies used for radiological imaging analysis reflect that CAD system upgrades the accuracy of diagnosis, eases the ever increasing workload and minimizes inter- and intra-observer variation [3]. The standard MR protocols used more often are: T1- weighted image (T1WI), proton density-weighted image (PDWI), T2-weighted image (T2WI), and T1WI after the administration of paramagnetic contrast enhancement agent such as Gadolinium (Gd). The T1 and T2 relaxation times for most of the brain tumors are much longer than the normal brain tissue. The tumors appear as hypo-intense portion of image on T1WI and hyper-intense on T2WI, compared to normal brain tissue intensity [4]. The CAD systems are used to provide a useful preliminary opinion for radiologists using various machine learning techniques. The presence of edema around the tumor and the heterogeneous nature of different types of tumor make the segmentation process a bit difficult and complex. Therefore the primary objective is to develop an efficient and accurate brain tumor detection system.

2 Related Work

Smith [5] developed an automated, fast and robust method for segmenting MR head scans into brain and non-brain parts. Somasundaram [6] proposed two unsupervised and knowledge based methods to extract brain parts automatically using region labelling and morphological operations.

A hybrid level set method for brain extraction was proposed by Jiang [7]. This method uses a non-linear speed function In unsupervised method, there is no need to label the image manually one by one or to select the number of classes, instead a specific algorithmic procedure is run to group the closest homogeneous regions of interest [8].

At present, techniques such as thresholding [9], region growing [10, 11], active contour and shape based models [4, 12, 13], knowledge based approaches [14, 15], machine learning based such as clustering [16–18] have been used for image segmentation. Otsu's Method [9] is used to select global threshold value from gray-level histogram. The optimal threshold value is calculated by maximizing the between class variance [19].

Selvakumar [20] proposed a brain tumor segmentation technique combining both K-Means and Fuzzy C-Means clustering algorithm. Ahmed [21] proposed an efficient technique using various morphological operations, wavelet transform and K-means clustering. A hybrid and accurate brain tumor segmentation technique [22] uses a combination of K-means clustering, Fuzzy C-means algorithm, thresholding and level set method.

Chaplot [27] proposed a brain MR image classifier using wavelet features to a neural network. Zhang [24] proposed a classifier which includes wavelet based features, principle component analysis (PCA) for reduction of feature dimension. Joshi [25] developed a Neuro Fuzzy classifier using Gray Level Co-occurrence Matrix (GLCM) for classification of different types of brain tumors.

3 The Proposed Methodology

The proposed methodology is aimed to develop a computer aided diagnostic system for automatic detection of brain tumor through MRI. The proposed CAD system is shown in Fig. 1.

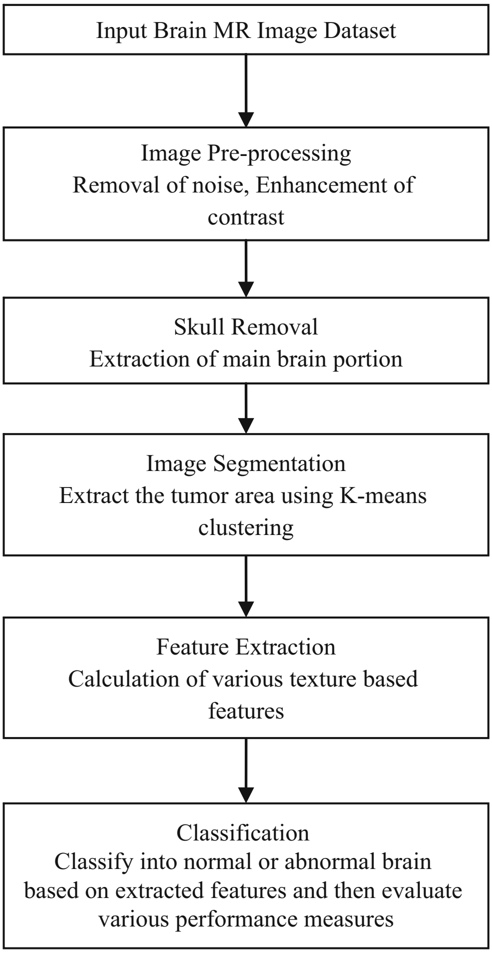

Input Brain MR Image Dataset
Image Pre-processing
Removal of noise, Enhancement of contrast
Skull Removal
Extraction of main brain portion
Image Segmentation
Extract the tumor area using K-means clustering
Feature Extraction
Calculation of various texture based features
Classification
Classify into normal or abnormal brain based on extracted features and then evaluate various performance measures

Fig. 1. Proposed CAD system for automatic detection of brain tumor

3.1 Input Brain MR Image Dataset

In this CAD system, the input images are taken as MR images as they provide rich information about soft anatomical tissue and have drastically enhanced the quality of brain pathological diagnosis and therapy [23].

The input image dataset contains 50 abnormal (images contain brain tumor) axial, T2-weighted MR images. The abnormal brain images contain various brain tumor types, such as Glioma, Sarcoma, Meningioma, Metastatic bronchogenic carcinoma, Metastatic adenocarcinoma. The images have dimension of 256 × 256 pixel and are taken from Harvard Medical School database [26].

3.2 Image Pre-processing

Image enhancement refers to various types of manipulation on the image matrix, resulting in an optimized output image. An automatic detection requires pre-processed images. Image pre-processing steps make the image segmentation more accurate by removing the common noise and artifacts and adjusting the contrast.

3.2.1 Removal of Noise

The noise is introduced while acquiring the image due to various hardware related artifacts, different environmental conditions, patient related motion artifacts because of gross movement of the body during the acquisition period and many others which leads to a degraded MR acquired signal. So the purpose is to eliminate these to a certain extent. While performing various image processing techniques the most common cause of noise encountered is due to intensity in homogeneity which makes segmentation process a bit difficult. To overcome this median filter is used for removal of impulse type (commonly known as salt and pepper noise) of noise. The main advantage of using median filter is that it preserves the edges while removing noise and gives a smoothed image. It improves the overall quality of the image [3].

3.2.2 Enhancement of Contrast

In medical imaging contrast is defined as the relative difference between the intensities of two adjacent regions. The factors that influence contrast in MR imaging includes magnetic field strength, magnetic field inhomogeneity, poor illumination and low sensitivity of neuroimaging sensors. While analysing the brain MR images it is found difficult to differentiate various tissue regions due to low contrast of the image. So the enhancement of contrast is necessary for those types of images. In addition various image segmentation procedures demand a good contrast enhanced image as input. The contrast is enhanced using a gray-level transformation called contrast stretching. In this proposed method the linear stretching of contrast is used.

3.2.3 Removal of Skull and Other Brain Tissues

In this work a fully-automatic segmentation approach is used which successfully separates main brain from non-brain regions. For this purpose the initial segmentation is carried out using global thresholding method proposed by Otsu [9]. The objective of this stage of segmentation is to generate a rough brain mask and then extract main brain portion.

3.3 Segmentation of Brain Tumor Using K-Means Clustering

Segmentation is the foremost image processing technique to separate the object of interest from a highly complex MR brain image. Clustering is a segmentation technique which assembles data to specific clusters with respect to some predefined criteria. This technique divides a set of data points into non-overlapping clusters of pixels based on intra-class equivalence. Objects belong to same cluster possess maximum amount of resemblance, but are heterogeneous to objects within different clusters [27]. All data points are considered at a time and similar type of data points are added to same cluster. Also each of these data points belonging same cluster exhibits almost same type of attributes.

In this proposed method, the K-Means clustering technique is utilized, which is an unsupervised data clustering method. This method groups objects into k number of groups based on their attributes. At first the distances between data points and their corresponding cluster centroid is calculated. Then the grouping is performed by finding the minimum Euclidean distance. As there are k numbers of clusters so there will be k number of initial cluster centroids.

3.4 Texture Feature Extraction Using Gray Level Co-occurrence Matrix

The next process after image segmentation is to extract the texture features from the segmented image which is found very helpful for proper analysis of image in a large scale. Texture is considered as a surface property of every object and each object has its own statistical textural features, so it differs from each other.

Haralick [28] first presented the gray-level co-occurrence matrices (GLCM) to extract different textural features. It uses 2-D histogram of different gray levels. In this approach two adjacent pixels are considered according to a specific spatial relationship. The co-occurrence of probabilities is calculated with the help of a displacement vector d and its orientation θ. In this work, the four angular orientations are taken as 0°, 45°, 90° and 135°. The separation distance d is taken as unity.

The feature vector includes three features derived from first-order statistics such as: (i) mean, (ii) standard deviation, and (iii) entropy; and from second-order statistics: (i) contrast, (ii) angular second moment, (iii) inverse difference moment, (iv) autocorrelation, (v) homogeneity, (vi) variance, (vii) cluster shade, (viii) sum average, (ix) inertia, (x) cluster prominence and (xi) dissimilarity [28].

3.5 Classification of Brain MR Images Using Support Vector Machine (SVM)

After extracting the valuable textural features, the brain MR images are classified into normal and abnormal classes using support vector machine (SVM). In this work, a two class SVM classifier is used. It takes labelled data as input from these two classes. Like other common machine learning techniques, SVM performs two fundamental steps such as, training and testing. The SVM gets its intelligence from its training set while classifying the unknown test dataset [23].

To apply SVM classifier, the input vectors are mapped to relatively high dimension feature space using linear or non-linear mapping technique and the separating hyperplane is constructed that has maximum distance from the closest points of the training set.

4 The Results and Discussion

The original images collected from Harvard Medical School database [26] and contain some amount of noise. This noise is removed using median filter. The original database images contain the brain with skull and other non-brain parts such as scalp, skin, eyeballs etc. These portions of the image are removed using several image processing steps.

After removing the skull and other non-brain parts the brain image is then segmented to extract the tumor part. In this work, different types of brain tumor images are segmented using K-means clustering technique with number of clusters taken as four. It is observed that cluster 4 contains the required tumor portion. The clustered images obtained are found to be the binary clustered images. Then these clustered images are further processed with the help of binary connected component analysis to get tumor portion. Then these binary tumor portion images are applied to original images to get the actual gray-scale tumor portion. The results of the segmentation process and the final segmented tumors output are shown in Fig. 2.

The texture features are calculated from Gray Level Co-Occurrence Matrices in this work. The texture features considered in this work are: autocorrelation, contrast, correlation, dissimilarity, energy, entropy, homogeneity, maximum probability, variance, sum average, sum variance, sum entropy, difference entropy, cluster prominence, cluster shade.

Table 1. SVM classification performance results

Iteration No.	Accuracy	Sensitivity	Specificity	Average accuracy	Average sensitivity	Average specificity
1	1	1	1	0.9928	1	0.9890
2	0.90625	1	0.85714			
3	1	1	1			
4	1	1	1			
5	1	1	1			
6	1	1	1			
7	1	1	1			
8	1	1	1			
9	1	1	1			
10	1	1	1			
11	1	1	1			
12	1	1	1			
13	1	1	1			

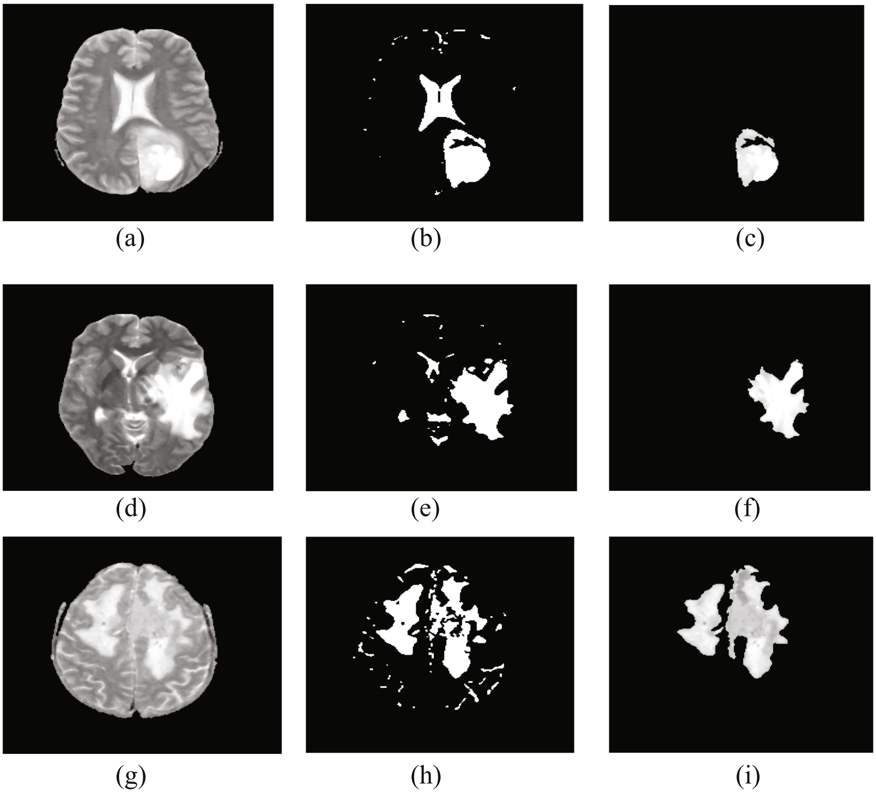


Fig. 2. K-Means segmentation of different types of brain tumor MR images: original skull-removed median filtered images (a, d, g), corresponding clustered images (b, e, h), segmented tumor (c, f, i).

The brain MR images are classified into normal and abnormal types. The input image dataset contains 22 normal brain images and 42 abnormal brain images having different types of brain tumor. In this method linear SVM classifier is used. Among the input data 50% of input images are used to train the classifier and based on this knowledge the other 50% data are tested.

These three measures are calculated over 13 iterations and the corresponding experimental results for normal and abnormal classification are listed in Table 1.

5 Conclusion

Many of the current segmentation techniques are operated on MR images due to its excellent soft-tissue contrast. The purpose of these methods is to assist the physicians to take a preliminary decision on diagnosis, monitoring, and therapy planning for the patients having brain tumor. The semi-automatic and fully automatic segmentation of brain tumor from abnormal brain images experiences great challenges due to irregularities in boundaries with discontinuities and partial-volume effects of brain tumor images. It is observed that the proposed methodology performs complete segmentation of the brain portion from non-brain tissues, skull and more importantly the tumor part. It requires only minimal manual selection of some parameters. Intra-observer variability which is often introduced in manual segmentation process is significantly reduced with the help of this automated method. Besides, due to the high amount of scans to be visualized, expert's fatigue, and the complexity of images, it is observed that some small brain tumor portions are being missed without using a computer-aided tool. Hence there is a risk of patient being not diagnosed properly. The utilization of a CAD tool would serve to improve not only the efficiency, but also the accuracy of brain tumor screening performed on cancer patients. The segmentation and classification accuracy with this automated method is very high. So the proposed system will definitely reduce the false positive and false negative rate and the cancer patients will be diagnosed in a more accurate way.

References

1. The Essential Guide to Brain Tumors, National Brain Tumor Society. http://www.braintumor.org
2. Drevelegas, A., Papanikolaou, N.: Imaging of Brain Tumors with Histological Correlations, pp. 13–18. Springer, Heidelberg (2011)
3. El-Dahshan, E.-S.A., Mohsen, H.M., Revett, K., Salem, A.-B.M.: Computer-aided diagnosis of human brain tumor through MRI: a survey and a new algorithm. Expert Syst. Appl. **41**, 5526–5545 (2014)
4. Sachdeva, J., Kumar, V., Gupta, I., Khandelwal, N., Ahuja, C.K.: A novel content-based active contour model for brain tumor segmentation. Magn. Reson. Imaging **30**, 694–715 (2012)
5. Smith, S.M.: Fast robust automated brain extraction. Hum. Brain Mapp. **17**, 143–155 (2002)
6. Somasundaram, K., Kalaiselvi, T.: Automatic brain extraction methods for T1 magnetic resonance images using region labeling and morphological operations. Comput. Biol. Med. **41**, 716–725 (2011)
7. Jiang, S., Zhang, W., Wang, Y., Zhen, C.: Brain extraction from cerebral MRI volume using a hybrid level set based active contour neighborhood model. Biomed. Eng. OnLine **12**, 31 (2013). http://www.biomedical-engineering-online.com/content/12/1/31
8. Gordillo, N., Montseny, E., Sobrevilla, P.: State of the art survey on MRI brain tumor segmentation. Magn. Reson. Imaging **31**, 1426–1438 (2013)
9. Otsu, N.: A threshold selection method from gray-level histograms. IEEE Trans. Syst. Man Cybern. SMC **9**(1), 62–66 (1979)
10. Dubey, R.B., Hanmandlu, M., Gupta, S.K.: Region growing for MRI brain tumor volume analysis. Indian J. Sci. Technol. **2**(9), 26–31 (2009)
11. Jafari, M., Kasaei, S.: Automatic brain tissue detection in MRI images using seeded region growing segmentation and neural network classification. Aust. J. Basic Appl. Sci. **5**(8), 1066–1079 (2011)
12. Li, C., Huang, R., Ding, Z., Chris Gatenby, J., Metaxas, D.N., Gore, J.C.: A level set method for image segmentation in the presence of intensity inhomogeneities with application to MRI. IEEE Trans. Image Process. **20**(7), 2007–2016 (2011)

13. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: active contour models. Int. J. Comput. Vis. **1**, 321–331 (1988)
14. Clark, M.C., Hall, L.O., Goldgof, D.B., Velthuizen, R., Reed Murtagh, F., Silbiger, M.S.: Automatic tumor segmentation using knowledge-based techniques. IEEE Trans. Med. Imaging **17**(2), 187–201 (1998)
15. Wagenknecht, G., Kops, E.R., Tellmann, L., Herzog, H.: Knowledge-based segmentation of attenuation-relevant regions of the head in T1-weighted MR Images for attenuation correction in MR/PET systems. In: IEEE Nuclear Science Symposium Conference Record, M09-287 (2009)
16. Portela, N.M., Cavalcanti, G.D.C., Ren, T.I.: Semi-supervised clustering for MR brain image segmentation. Expert Syst. Appl. **41**, 1492–1497 (2014)
17. Agrawal, S., Panda, R., Dora, L.: A study on fuzzy clustering for magnetic resonance brain image segmentation using soft computing approaches. Appl. Soft Comput. **24**, 522–533 (2014)
18. Jude Hemanth, D., Selvathi, D., Anitha, J.: Effective fuzzy clustering algorithm for abnormal MR brain image segmentation. In: IEEE International Advance Computing Conference (IACC 2009), Patiala, India, 6–7 March 2009
19. Moallem, P., Razmjooy, N.: Optimal threshold computing in automatic image thresholding using adaptive particle swarm optimization. J. Appl. Res. Technol. **10**, 703–712 (2012)
20. Selvakumar, J., Lakshmi, A., Arivoli, T.: Brain tumor segmentation and its area calculation in brain MR images using K-mean clustering and fuzzy C-mean algorithm. In: IEEE-International Conference on Advances in Engineering, Science and Management, March 2012
21. Kharrat, A., Ben Messaoud, M., Benamrane, N., Abid, M.: Detection of brain tumor in medical images. In: International Conference on Signals, Circuits and Systems (2009)
22. Abdel-Maksoud, E., Elmogy, M., Al-Awadi, R.: Brain tumor segmentation based on a hybrid clustering technique. Egypt. Inform. J. **16**, 71–81 (2015)
23. Chaplot, S., Patnaik, L.M., Jagannathan, N.R.: Classification of magnetic resonance brain images using wavelets as input to support vector machine and neural network. Biomed. Signal Process. Control **1**, 86–92 (2006)
24. Zhang, Y., Wu, L.: An MR brain images classifier via principal component analysis and kernel support vector machine. Prog. Electromagn. Res. **130**, 369–388 (2012)
25. Joshi, D.M., Rana, N.K., Misra, V.M.: Classification of brain cancer using artificial neural network. In: 2nd International Conference on Electronic Computer Technology (2010)
26. http://med.harvard.edu/ANNALIB/
27. Rahmani, M.K.I., Pal, N., Arora, K.: Clustering of image data using K-means and fuzzy K-means. Int. J. Adv. Comput. Sci. Appl. **5**(7), 160–163 (2014)
28. Harlick, R.M., Shanmugam, K., Dinstein, I.: Textural features for image classification. IEEE Trans. Syst. Man Cybern. SMC **3**(6), 610–621 (1973)

Feature Selection Using Genetic Algorithm for Big Data

Rania Saidi(✉), Waad Bouaguel Ncir, and Nadia Essoussi

LARODEC, ISG, University of Tunis, Tunis, Tunisia
rania.saidi28@gmail.com, bouaguelwaad@mailpost.tn,
nadia.essoussi@isg.rnu.tn

Abstract. Feature selection is a powerful technique for dimensionality reduction and an important step in successful machine learning applications. In the last few decades, data has become progressively larger in both numbers of instances and features which make it harder to deal with the feature selection problem. To cope with this new epoch of big data, new techniques need to be developed for addressing this problem effectively. Nonetheless, the suitability of current feature selection algorithms is extremely downgraded and are inapplicable, when data size exceeds hundreds of gigabytes. In this paper, we introduce a scalable implementation of a parallel feature selection approach using the genetic algorithm that has been done in parallel using MapReduce model. The experimental results showed that the proposed method can be suitable to improve the performance of feature selection.

Keywords: Feature selection · Genetic algorithm · MapReduce
Parallel computing · Big Data

1 Introduction

The huge amount of data is a challenging issue that demands a large computational infrastructure to guarantee successful data processing and analysis.

Unfortunately, dimensionality comes with a problem since data has become progressively larger in both number of instances and number of features, in which existing algorithms of classification need to reduce the number of features in order to work efficiently without a snag [5]. Thus, data reduction is reducing the number of initial features so as to select a subset that holds enough information and can best represent data in order to obtain good learning results.

There are two ways for reducing data: feature extraction and feature selection. In feature extraction, new attributes are generated from the initial ones while in feature selection a subset relevant features is selected without a transformation [5].

Genetic algorithms (GAs) are one of the most extensively used algorithms for feature selection since they can improve the performance of data mining algorithms. The main idea of this algorithm is analogous to the mechanism of

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 352–361, 2018.
https://doi.org/10.1007/978-3-319-74690-6_35

natural evolution in life sciences [6]. Genetic algorithms are considered as metaheuristics approaches that converge towards optimal or near optimal solutions. It provides a sufficiently good solution to any optimization problem which is, in our case, feature selection.

It starts with an initial population that is presented as a string (set of chromosomes). In each iteration, the algorithm generates a new population applying genetic operations on the selected individuals, such as crossover and mutation. Along several generations, the population aims to improve its solutions (individuals) by preserving the best individuals (using a fitness value) and eliminating the weakest ones. This process continues until a stopping criterion is reached. Using the best fitness value in the last population, the solution is given. In the case of a single memory execution, using a large population and processing intensive computing can slow down the process. Lately, the amount of data that is being produced and stored is nearly inconceivable, and it remains growing. It is renowned as Big Data: this term refers to the amount of data that is beyond traditional technology's ability to perform the storage, managing and processing successfully. Besides, the growing numbers of instances and features outlines new challenges for machine learning algorithms, such as feature selection, that are commonly used to process a small amount of data. To the best of our knowledge, most existing studies of Genetic algorithm based feature selection are limited to small-scale data bases and have to be adapted to deal with Big Data.

In view of this requirements, attempts have been made to parallelize GA to prevent scalability issues using various frameworks and techniques. Parallel systems are widely used due to the increasing popularity of Cloud systems and distributed platforms such as Apache Hadoop, characterized by its easy scalability. The Apache Hadoop ecosystem is eventually well-known due to the MapReduce model and Hadoop Distributed File System HDFS that runs on large clusters of commodity machines. Actually, GA can be parallelized within three models: Global parallelization model, grid model, and island model. But, one of the problems in applying GAs in a distributed system is the overhead. Therefore, the aim of this work is to propose a method for adapting island GA task by processing a large population on the Hadoop platform using the MapReduce paradigm. This method consists of three main steps: the vertical splitting, the map and the reduce. In the first step, the dataset is splitted vertically according to a fixed number of features into smaller subsets in order to imitate the island model. Then, the genetic algorithm based features selection is applied in each subset in the map step. Finally, the reduce is achieved by merging the final selected features.

The rest of the paper is organized as follows. Section 2 presents some background about feature selection algorithms using the MapReduce programming model and the state of the art about genetic algorithm using MapReduce. Section 3 describes the proposed approach for the Genetic algorithm using the MapReduce. The results are presented and analyzed in Sect. 4 Finally the summary of the work is afforded in the conclusion in Sect. 5

2 Related Work

Many approaches have been proposed to deal with feature selection problem. But, many of them don't cope with large dimensionality that refers to big data term. This latter refers to a huge amount of data that exceeds the capacity of memory and disk [1]. Big data can be characterized basically by three dimensions: volume, variety, and velocity [9].

Despite being studied extensively, most existing feature selection works are restricted to small-scale data bases, which assumes that the feature selection task is conducted on small databases where all data are centralized on a single machine. Such assumptions may not always hold for real-world applications in which the number of features is high.

According to this reason, many distributed frameworks have been suggested in literature: Message Passing (MPI), threads, workflow and MapReduce. Indeed, parallel systems are extensively used due to the increasing popularity of Cloud Systems and the availability of distributed platforms, such as Apache Hadoop. Many researchers have used the latter one to handle big data problems. In fact, working with Hadoop Distributed File System (HDFS) and MapReduce (a model of distributed programming) affords an ideal environment to implement parallel and distributed solutions for massive data applications. The MapReduce model have three main steps: a map step, a shuffle step and a reduce step as presented in Fig. 1. Nonetheless, the map function and the reduce are programmed to perform a variety of tasks by users but the Shuffle is executed automatically by [9]. MapReduce starts by splitting the input into several independent blocks. Then, the Mapper starts processing them in a parallel manner, each block with a map. The output of the map phase is a set of (Key, value) pairs. After that, the Shuffle collects all pairs together and groups them by the key automatically.

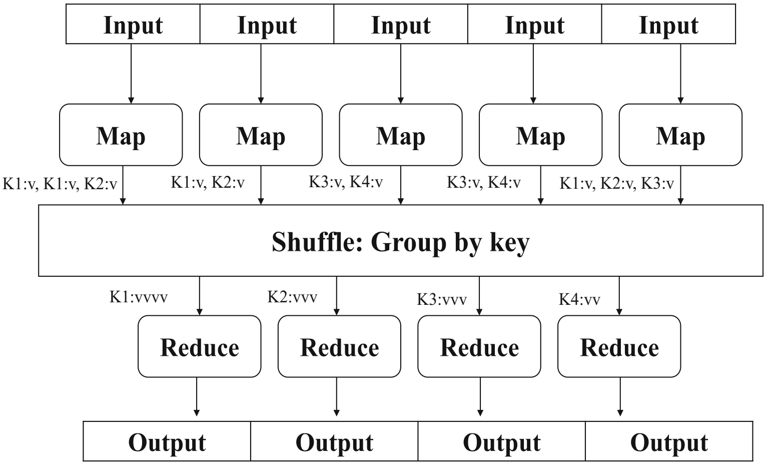


Fig. 1. MapReduce model

Next, the Reducer action is applied to the output of the Shuffle phase. Each key is assigned to a reduce and the outputs of the Reducer are collected as the desired result [8].

As described by [3], there exists three possible ways to parallelize GAs using the Mapreduce paradigm: Global parallelization model, Fine-grained parallelization model or grid model and Coarse-grained or island model. In the first one, a "master" node applies genetic and selection operators and the remaining nodes, called "slave", compute the fitness values of the individuals.

In the second model, each individual is located on a grid (a node) and all GA operations are performed in parallel evaluating simultaneously the fitness and applying locally selection and genetic operations to a small neighboring [3].

In the third model, the population is divided into different subpopulations of broad size which are placed in several nodes (islands). Actually, the Genetic algorithm is applied on each subpopulation. Then, an exchange of information is performed by allowing some individuals to migrate from one island to another according to some given criteria [3]. [4] developed a framework for parallel Genetic Algorithms (GAs) on the Hadoop platform by following the MapReduce model. The framework focuses on the aspects of GA that are specific to the problem to be addressed following the island model. This framework has been devised to address the Feature Subset Selection problem.

[2] introduced a scalable implementation of a parallel genetic algorithm in Hadoop MapReduce using the rough set for subset selection. The genetic operators took place in the reduce phase.

In [7], the problem of feature selection has been considered where GA has been found an effective technique for searching for quality solution. To improve the performance they used master slave Parallel Genetic algorithm using Hadoop MapReduce and they suggested a wrapper technique using KNN classifier for supervised feature selection.

All these works have shown good results but they do not deal with high dimensionality.

3 Proposed Approach: Genetic Algorithm Using the MapReduce Model for Feature Selection (MR-GAFS)

In order to make possible the process of feature selection for a huge and massive amount of data, we design a parallel method within the MapReduce model. Hence, Genetic algorithm was the chosen algorithm since it gives good learning results [6].

The proposed method aims to exploit the features of Hadoop in terms of scalability in an ideal infrastructure for a distributed computation to handle large datasets.

To overcome the overhead that has been mentioned before, the proposed method implements the island model by splitting the datasets into several subsets called islands on which we apply the genetic algorithm.

In the first step, the big number of features is splitted into several subsets of features. Therefore, many smaller datasets are generated from the original one by a vertical splitting of the big dataset.

Then, for the map step, the resulting subsets are the input value with an input key which refers to the name of the generated dataset. In this phase, the genetic algorithm for feature selection is applied giving the selected features and their values.

The map phase output, constitutes the input of the reduce phase in which all the relevant features are aggregated in one file. Hence, the large number of features is handled by the vertical splitting of the dataset. In addition, each file of the map step is splitted horizontally by default which allows to deal with the big number of instances (Fig. 2).

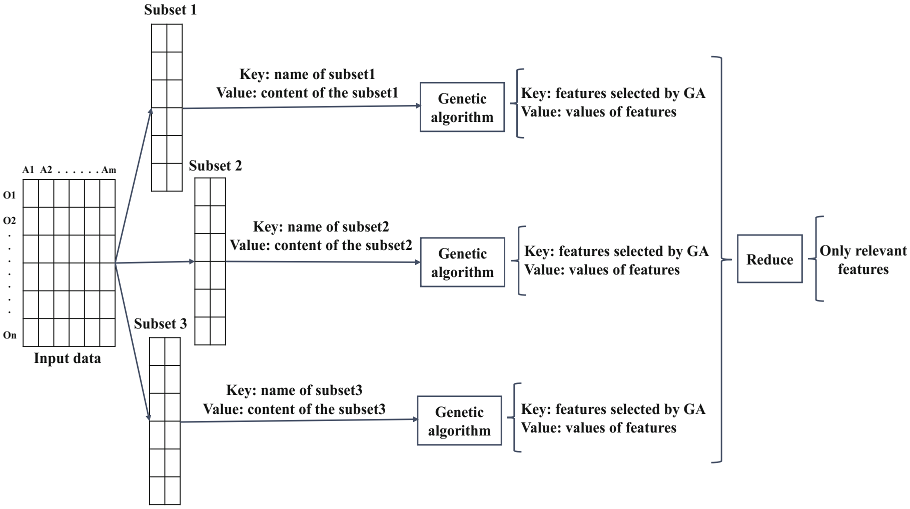

Subset 1
Subset 2
Subset 3
A1 A2 . . . . . . Am
O1
O2
On
Input data
Key: name of subset1
Value: content of the subset1
Key: name of subset2
Value: content of the subset2
Key: name of subset3
Value: content of the subset3
Genetic algorithm
Genetic algorithm
Genetic algorithm
Key: features selected by GA
Value: values of features
Key: features selected by GA
Value: values of features
Key: features selected by GA
Value: values of features
Reduce
Only relevant features

Fig. 2. Proposed approach

4 Results and Discussion

In order to assess the performance and effectiveness of our proposed approach, datasets from UCI machine learning repository were used. The first one is the Madelon dataset which was used in the Neural Information Processing Systems (NIPS) 2003 feature selection challenge. It is a two-class classification problem with 500 features and 4400 instances. We also used Semeion Handwritten Digit dataset that consists of 1593 instances and 256 features. Finally, we consider the BreastCancer dataset that consists of 569 instances and 32 features.

Our approach was implemented using the Hadoop MapReduce implementation with R language on Amazon Web Services Cloud. The RHadoop was used as an open source project that provides several R packages to work with R and

Hadoop interactively. It associates R with Hadoop to link R's statistical effectiveness with the scalable compute power given by Amazon Elastic MapReduce (EMR) on the Hadoop MapReduce model. This mixture allows processing a large amount of data on Amazon EMR which otherwise would not be possible using R in stand-alone mode.

As mentioned before we deal with three kinds of implementation: a sequential, a Hadoop pseudo-distributed and a full distributed version on Amazon cluster. The sequential and the Hadoop pseudo-distributed versions were executed on a single machine with 6 GB of RAM and the full distributed one was implemented over a Hadoop platform. The full distributed version required a full Hadoop cluster composed of 1 master and 3 slaves. These nodes are named by Amazon as "m4.large", each machine having 2 CPUs and 8 GB of RAM. Table 1 illustrates the experimental environment for each version.

Table 1. Experimental environment

	Sequential	Hadoop pseudo-distributed	Full distributed
Number of machines	One machine	One machine	Master/3 slaves
RAM	6 GB	6 GB	8 GB
Framework	_	Hadoop 2.7.3	Hadoop 2.7.3
Language	R (3.3.1)	R (3.3.1)	R (3.3.1)

As shown in Table 2, during the experimental phase we stimulated different values of parameters, then we chose the best values that give the best accuracy in order to use it in the rest of experiments. Thereby, the chosen parameters for the test consisted of 100 generations of populations with 20 individuals each.

Table 2. Parameters setting of Genetic algorithm

Number of generations	Population size	Crossover rate	Mutation rate	Accuracy
10	10	0.8	0.1	0.4589
30	30	0.8	0.1	0.4617
50	20	0.7	0.1	0.4622
50	50	0.8	0.1	0.4599
100	20	0.8	0.1	0.4889

Random forests model and 10 fold cross validation are used to assess performance of the "chromosomes" in each generation. Random Forest consists in growing an ensemble of trees and then, combine those trees predictors by majority vote. In fact, it is a combination of classifiers, made by aggregating the predictions of the ensemble to make a final prediction. It consists essentially on generating a specific number of trees, to let them later vote for the most popular class. Indeed, a feature selection occurs into the random forests algorithm by selecting features that improve most the predictive performance to place them in the tree nodes. Under those facts, random forests are treated as an embedded feature selection method which produces a high prediction accuracy.

In Genetic Algorithm using Random Forest and for each generation, individuals are used to produce a forest of decision trees. Then a fitness score is assigned to each individual based on how well the corresponding tree classifier classified the test dataset using RMSE.

To evaluate our proposed approach, two categories of evaluation measure are considered. The first category evaluates the performance in terms of running time and the second one evaluates the influence of our approach on the learning step. Thus, we use some classification measures: Accuracy and F-measure.

The given results in Tables 3 and 4 show that the running time of Genetic algorithm using the proposed approach decreases compared to the nonparallel version. It varies from 207945.992 to 151949.403 for Madelon dataset and from 28490.67 to 16428.100 for Semeion Handwritten Digit dataset. This improvement is due to increasing number of nodes in the full distributed version. Literally, when using MapReduce: increasing the number of nodes decreases the running time. The dissimilarity in the number of selected features for the GA is explained by the randomness in evaluating the features. For example, the number of selected features is between 238 and 261 for Madelon dataset.

Table 3. Results given by Madelon dataset

	Running time (s)	Number of selected features
Non Parallel (Sequential)	207945.992	261
Parallel (Pseudo-distributed)	204530.196	238
Parallel (Full-distributed)	151949.403	246

Table 4. Results given by Semeion Handwritten Digit dataset

	Running time (s)	Number of selected features
Non Parallel (Sequential)	28490.67	81
Parallel (Pseudo-distributed)	26954.098	110
Parallel (Full-distributed)	16428.100	114

In Tables 5, 6, 7, and 8 we evaluate the impact of our proposed feature selection method on the learning step using Support Vector Machine classifier and RandomForest classifier. The obtained results show the effectiveness of selecting features using the island model based MapReduce in improving the classification results. In fact, the large number of features can downgrade some learning algorithms leading to long training time. Support Vector Machines are particularly

Table 5. Classification results for Madelon dataset using Support Vector Machine

	Time for building the model (s)	Accuracy	F-measure
All features	2.440	0.4855	0.58
Sequential (261 features)	0.408	0.4889	0.57
Pseudo-distributed (238 features)	0.372	0.5206	0.581
Full-distributed (246 features)	0.381	0.5331	0.59

Table 6. Classification results for Madelon dataset using RandomForest

	Time for building the model (s)	Accuracy	F-measure
All features	6.608	0.5111	0.56
Sequential (261 features)	3.386	0.4830	0.59
Pseudo-distributed (238 features)	3.088	0.5145	0.61
Full-distributed (246 features)	3.262	0.5261	0.61

Table 7. Classification results for Semeion Handwritten Digit dataset using SVM

	Time for building the model (s)	Accuracy	F-measure
All features	0.592	0.9790	0.98
Sequential (81 features)	0.168	0.9796	0.98
Pseudo-distributed (110 features)	0.348	0.9916	0.99
Full-distributed (114 features)	0.398	0.9920	0.99

Table 8. Classification results for Semeion Handwritten Digit dataset using RF

	Time for building the model (s)	Accuracy	F-measure
All features	716.608	0.9727	0.98
Sequential (81 features)	226.004	0.9748	0.98
Pseudo-distributed (110 features)	3.528	1	1
Full-distributed (114 features)	3.630	1	1

well suited to this case. SVMs have the ability to separate classes more quickly and with less overfitting than most other algorithms by using a modest amount of memory.

For example, when using the SVM classifier for the Madelon dataset, the accuracy and the F-measure without selecting features were 0.4855 and 0.58 respectively, while they were about 0.5331 and 0.59 for our proposed method. These two latters values are higher than those provided by the sequential and the pseudo-distributed version.

Tables 5, 6, 7, and 8 show that when a classification algorithm is applicable for all features, selecting features using MapReduce reduces the time for building models. For Madelon dataset, it decreases from 0.408 s to 0.372 s using Support Vector Machine and decreases from 6.608 to 3.088 using RandomForest model. Besides, our proposed approach enhances the Accuracy and the F-measure.

To highlight more our method we compare it to other existing ones. The first one consists in a fuzzy GA using SVM as a fitness function (FPIMMOGA) [10]. In the latter one, the parallelization was performed by the Open MPI, that allows to work with many computers by passing messages, and by implementing the island model. The second one consists in using a filter measure, which is the consistency, to select relevant features [11]. We tested the accuracy of our method using Breast cancer datasets comparing to other approach. Actually, the accuracy results given by our method are better than those given by the GA based SVM method and the Consistency method. Table 9 details the results.

Table 9. Compative analysis

	MR-GAFS	FPIMMOGA	Consistency
Accuracy using RF	0.9569	0.945	0.926
Accuracy using SVM	0.9717	0.965	0.9764

5 Conclusion

In this work we proposed a new approach dealing with Big datasets using MapReduce that consists in a parallel implementation of the Genetic algorithm for feature selection problem. Experimental results showed that the idea is suitable and presents good results for large datasets. Besides, we proved that we can improve the quality of classification algorithms on high dimensional data. For future work, we intent to experiment our approach on massive data by enlarging the number of nodes and we can also study missing values and noisy data in order to enhance learning results.

References

1. Cox, M., Ellsworth, D.: Application-controlled demand paging for out-of-core visualization. In: Proceedings of the 8th Conference on Visualization, 1997, p. 235-ff. IEEE Computer Society Press (1997)
2. Di Geronimo, L., Ferrucci, F., Murolo, A., Sarro, F.: A parallel genetic algorithm based on hadoop mapreduce for the automatic generation of junit test suites. In: Software Testing, Verification and Validation (ICST), IEEE Fifth International Conference, pp. 785–793. IEEE (2012)
3. El-Alfy, E.S.M., Alshammari, M.A.: Towards scalable rough set based attribute subset selection for intrusion detection using parallel genetic algorithm in mapreduce. Simul. Model. Pract. Theory **64**, 18–29 (2016)
4. Ferrucci, F., Salza, P., Kechadi, M., Sarro, F.: A parallel genetic algorithms framework based on Hadoop MapReduce. In: Proceedings of the 30th Annual ACM Symposium on Applied Computing, pp. 1664–1667 (2015)
5. Garca, S., Luengo, J., Herrera, F.: Data Preprocessing in Data Mining, pp. 59–139. Springer, New York (2015)

6. Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, Reading (1989)
7. Hilda, G.T., Rajalaxmi, R.R.: Effective feature selection for supervised learning using genetic algorithm. In: Electronics and Communication Systems (ICECS), 2nd International Conference IEEE, pp. 909–914 (2015)
8. Kacem, M.A.B.H., N'cir, C.E.B., Essoussi, N.: MapReduce-based k-prototypes clustering method for big data. In: Data Science and Advanced Analytics (DSAA). 36678 2015. IEEE International Conference, pp. 1–7. IEEE(2015)
9. Sagiroglu, S., Sinanc, D.: Big data: a review. In: Collaboration Technologies and Systems (CTS), 2013 International Conference IEEE, pp. 42–47 (2013)
10. Natarajan, A., Balasubramanian, R.: A fuzzy parallel island model multi objective genetic algorithm gene feature selection for microarray classification. Int. J. Appl. Eng. Res. **11**(4), 2761–2770 (2016)
11. Yu, L., Liu, H.: Efficient feature selection via analysis of relevance and redundancy. J. Mach. Learn. Res. **5**, 1205–1224 (2004)

Prediction of Liver Diseases Based on Machine Learning Technique for Big Data

Engy A. El-Shafeiy[1](✉), Ali I. El-Desouky[1], and Sally M. Elghamrawy[2]

[1] Computers and Systems Department, Faculty of Engineering,
Mansoura University, Mansoura, Egypt
engy.elshafeiy@gmail.com

[2] Computers Engineering Department, MISR Higher institute for Engineering
and Technology, Al Mansora, Egypt
sally_elghamrawy@ieee.org

Abstract. Liver diseases have produced a big data such as metabolomics analyses, electronic health records, and report including patient medical information, and disorders. However, these data must be analyzed and integrated if they are to produce models about physiological mechanisms of pathogenesis. We use machine learning based on classifier for big datasets in the fields of liver to Predict and therapeutic discovery. A dataset was developed with twenty three attributes that include the records of 7000 patients in which 5295 patients were male and rests were female. Support Vector Machine (SVM), Boosted C5.0, and Naive Bayes (NB), data mining techniques are used with the proposed model for the prediction of liver diseases. The performance of these classifier techniques are evaluated with accuracy, sensitivity, specificity.

Keywords: Classifier techniques · Machine learning · Liver diseases
Data mining · Big data

1 Introduction

According to the WHO in Egypt the chronic diseases are responsible for 78% (382,000/495,000) of total deaths and in next 10 years twenty lakhs of people will die due to chronic diseases [1]. Liver diseases also come in the category of chronic diseases. A large number of infections affect the liver which led to the various liver diseases. The deaths due to the liver diseases have reached 208185 or 7.34% of total death in Egypt [2]. Liver is one of the strongest organs in our body that sits on the right side of the belly. Color of the liver is reddish-brown. The liver has two sections in our body, one is called right section and the other is left section. The place of gallbladder is under the liver, alongside parts of the pancreas and guts. The liver and these organs cooperate to assimilate and process sustenance. Liver sanitizes the blood originating from the digestive tract. Likewise, cleans chemicals and metabolizes drugs. The liver shrouds bile that winds up back in the inner parts. The liver additionally makes proteins imperative for blood coagulating. The liver is in charge of the evacuation of pathogens and exogenous antigens from the systemic flow. Liver infection is alluded to as hepatic ailment. Liver

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 362–374, 2018.
https://doi.org/10.1007/978-3-319-74690-6_36

infection is a general term that covers all the potential issues that bring about the liver to neglect to perform its assigned capacities. This study mainly discusses about five types of liver diseases such as alcoholic liver damage (ALD), liver cirrhosis (LC), primary hepatoma (PM), cholelithiasis (C) [3] and HCC [4].

One of the main causes of increased liver diseases in Egypt is obesity, inhale of harmful gases, intake of contaminated food, excessive consumption pickles and drugs, alcohol [2]. The objective of this paper is to propose a Machine learning techniques based on Classification of Liver disorders for reduce burden on doctors.

The organization of this paper is as follows. In Sect. 2, some related work on data mining, liver disease, classification algorithms, machine learning, and related works are provided. Section 3 describes our method in the implementation of Support Vector Machine (SVM), Boosted C5.0, and Naive Bayes (NB) classification algorithms for the early detection of liver diseases. Finally, in Sect. 4, Conclusion and future works.

2 Related Work

2.1 Machine Learning and Knowledge Discovery

Biomedical science is one of the important areas where data mining is used. Since this branch of science deals with human life, it is highly sensitivities. In recent years, a lot of researches have been done on a variety of diseases using data mining.

Looking more closely at the research done in recent years in this field, specifically, in the biomedical field, we can see many works that use data mining for forecasting, prevention and treatment of patients [5].

In biomedical science, accuracy and speed are two important factors that should be considered chiefly in dealing with any disease. In this regard, data mining techniques can be of great help to physicians.

With advances, several machines have entered in our lives. One of the most famous areas where computers as the mostly used machines can be helpful is knowledge extraction with the help of a machine (machine learning).

This approach that can be of great help to all scientific fields is called data mining orKnowledge Discovery of the Datasets as shown in Fig 1. Supervised and unsupervised learning are two main methods for machine learning [6]. The purpose of these methods is to learn by use of data mining approaches and to use part of data for training and the other part for test.

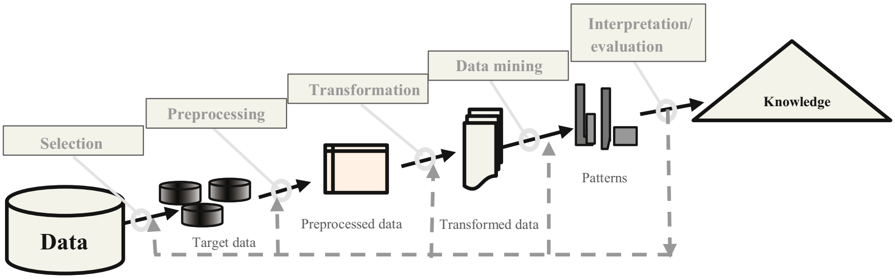


Fig. 1. The Data mining or Knowledge Discovery in Datasets (KDD) process [7].

2.2 Liver Diseases

Liver is largest internal organ of the body. It plays a significant role in transfer of blood throughout our body. The levels of most chemicals in our blood are regulated by the liver. It helps in metabolism of the alcohol, drugs and destroys toxic substances. Liver can be infected by parasites, viruses which cause inflammation and diminish its function [8]. It has the potential to maintain the customary function, even when a part of it is damaged. However, it is important to diagnose liver disease early which can increase the patient's survival rate. Expert physicians are required for various examination tests to diagnose the liver disease, but it cannot assure the correct diagnosis [9]. Accordingly, the mortality and morbidity impact of chronic liver disease is greatest among the population of Egypt.

2.3 Big Data

Big data turns into segments due to multidisciplinary combined effort of machine learning, datasets and statistics. Today, in biomedical sciences disease diagnostic test is a serious task [10]. In biomedical sciences disease diagnostic test is a serious task.

It is important to understand the exact diagnosis of patients by assessment and clinical examination. Medical field generates big data about report regarding patient, clinical assessment, cure, follow-ups, and medication. Enhancement in big data needs some proper means to extract and process data effectively and efficiently [11]. One of the many machine-learning is employed to build such classifier that can divide the data on the basis of their attributes. Dataset is divided into two or more than two classes. Such classifiers are used for medical big data analysis and disease detection.

2.4 Classification Algorithms

Naive Bayes classifier

Naive Bayes is a statistical classifier based on Bayes theorem for Thomas Bayes who worked in decision theory and probability [12]. Some literation mentioned that the Naive Bayes has simplicity, traceability and fast learner. On the other hand, many authors concentrated on class conditional independence assumption's advantages and disadvantages due to its influence on Naive Bayes performance whereas the Naive Bayes assumes that the attributes are independent among each other on a given class.

To compute the posterior probability that tuple X = (x_1, x_2, x_3,..x_n) belongs to the class C_i, we use the Eq. 1 below where xi is the value of attribute Ai and x_n is the value of attribute An.

$$P\left(\frac{C}{X}\right) = \frac{P\left(\frac{X}{C}\right)P(C)}{P(X)} \tag{1}$$

Where

P (c/x) is the posterior probability of class (target) given predictor (attribute).
P(c) is the prior probability of class.
P (x/c) is the likelihood which is the probability of predictor given class.
P(x) is the prior probability of predictor

C5.0 classifier

The decision tree such as C5.0, Id3, or CART can handle the real world datasets efficiently [13]. C5.0 is a Decision Tree was designed ID3 which is based on information gain. Because of the ID3 biases of multivalued attributes. C5.0 was designed to solve that problem by computing the information gain ration for each attribute then select the attribute has the maximal Information Gain ration value to be a root node of the training dataset. The attribute of the maximum gain ratio is picked up for splitting to reduce the needed information to predict a given instance in the resulting attribute's partition, the Gain Ration for attribute A is computed as follows:

$$GainRatio(A) = \frac{Gain(A)}{Split\,Info(A)} \tag{2}$$

$$Gain(A) = Info(D) - Info_{\mathrm{A}}(D) \tag{3}$$

Where D is the training dataset.

$$Info(D) = -\sum\nolimits_{i=1}^{n} p(ci)\log_2 P\left(C_{\mathrm{i}}\right) \tag{4}$$

$P(Ci) = |C_{\mathrm{i,D}}|/|\mathrm{D}|$ Where $\left|C_{\mathrm{i,D}}\right|$ is the number of the tuples of the class C_i in the training dataset and |D| is the number of the tuples of the training dataset, and n is the number of the class's values.

$$Info_{\mathrm{A}}(D) = \sum\nolimits_{i=1}^{n} \left(\frac{|\mathrm{a_{i,D}}|}{|D|}\right) \times \left(-\sum\nolimits_{j=1}^{m} \frac{|C_{\mathrm{JD}}|}{|\mathrm{a_{i,D}}|}\log_2 \frac{|\mathrm{C_{J,D}}|}{\mathrm{a_{i,D}}}\right) \tag{5}$$

Where $|a_{i,D}|$ is the number of the tuples of the value a_J of the attribute A in the training dataset and |D| is the tuples of the training dataset and n is the number of the values of attribute A. $|C_{J,D}|$ is the number of the tuples of class C_J related with the value a_i of the attribute A and m is the number of the classes of class C.

As mentioned above that Information Gain is biased for multivalued attributes for example, serial number attribute will get the maximal value in Eq. 3 but will be useless in classification stage. To avoid this bias, the gain value of attribute A is dived by a measure gives the potential information after splitting the training dataset into v datasets according to the values of attribute A. This measure is called split information which is used in information gain ratio in Eq. 2. The split information is defined in Eq. 6:

$$Split\ Info_{\mathrm{A}}(\mathrm{D}) = \sum\nolimits_{J=1}^{V} \frac{|D_{\mathrm{J}}|}{|D|} \times \log_2 \frac{|D_{\mathrm{J}}|}{|D|} \tag{6}$$

Where $|D_J|$ is the number of the tuples of the value of attribute A and $|D|$ is the number of the tuples of the training dataset. For continuous attributes, firstly, sort the values in ascending order, secondly, set a split point for each pair of adjacent values $\frac{a_i + a_{i+1}}{2}$ then compute the $Info_A(D)$ for each $a_1 <$ Split point and $a_2 >$ split point where a_1 represents all values before the split point and a_2 represents all values after the split point of the continuous attribute A in Eq. 6. Finally, the best split point is selected based on the minimal value of $Info_A(D)$ in Eq. 5.

Support Vector Machine

SVM is a supervised learning method used for both classification and regression. It has very high generalization performance. There is no requirement to add a prior knowledge, even when it has very high input space dimension. This makes it a very good quality classifier. The main intend of the SVM classifier is to discriminate between members of two classes in the training data by finding best classification function. SVM is a generalized linear classification method. It simultaneously maximizes the geometric margin and minimizes the classification error [14].

Viewing input data as two sets of vectors in n dimensional space, a separating hyperplane will be constructed by SVM which maximizes the margin between two data sets. Two parallel hyperplanes are constructed in order to calculate the margin, one on each side of separating hyperplane. The largest distance to the neighboring data points of both the classes helps to achieve good separation. If the margin is large then the generalization error will become less. Thus, the support vectors and margins help to find the hyperplanes. Here the data points are considered in the form of:

$$\{(z_1, x_1), (z_2, x_2), (z_3, x_3), (z_4, x_4) \ldots \ldots \ldots \ldots, (z_i,\ x_i)\}$$

Here $z_i = 1/-1$ is a constant which donates the class to which z_i belongs where i is the number of samples. z_i represents m-dimensional real vector. With the help of separating hyperplane the training data can be easily viewed, which is

$$u.z + c = 0,$$

Where u is m-dimensional vector and c is scalar.

There is a vector u which is perpendicular to the dividing hyperplane and the scalar parameter c helps to increase the margin. In case of absence of c the hyperplane is passed through the origin by force. In order to maximize the margin, parallel hyper-planes are required. These parallel hyperplanes are described by the equation.

$$u.z + c = 1$$
$$u.z + c = -1$$

If training data is separated linearly, the parallel hyperplane are chosen as there are no points among them. Here by geometrically, the distance between the hyper planes can be find which is 2/|u|. This is the reason why there is need to lessen the |u|. The equation is:

$$\mathrm{u.z_J} - \mathrm{c} \geq 1 \,\mathrm{or}\, \mathrm{u.z_J} {-} \mathrm{c} < -1 \tag{7}$$

To define formally a hyper plane the following notation can be used:

$$L(z) = \alpha_o{+}\alpha^c z \tag{8}$$

Where α weight is vector and α_o is the bias.

The optimal hyperplane can be represented by scaling of α and α_o. There is one representation has been chosen from different possible representations of hyperplane which is as follows:

$$\left|\alpha_o{+}\alpha^c z\right| = 1 \tag{9}$$

z represents the training examples that are close to the hyperplane and these are known as support vectors. This representation is also known as canonical hyperplane.

The following equation gives the distance between the point z and a hyperplane

$$(\alpha_o, \alpha){:}\, Distance = \frac{\left|\boldsymbol{\alpha_o}{+}\boldsymbol{\alpha^c}\mathrm{z}\right|}{\boldsymbol{\alpha}} \tag{10}$$

But for canonical hyperplane the numerator is equal to one and distance to the support vector is:

$$\mathrm{Distance}_{\text{support vector machine}} = \frac{\left|\boldsymbol{\alpha_o}{+}\boldsymbol{\alpha^c}z\right|}{\boldsymbol{\alpha}} = \frac{1}{\boldsymbol{\alpha}} \tag{11}$$

Here, margin which is used in the above, is denoted by M is twice the distance to the closest examples $M = \frac{2}{\alpha}$.

Here, the problem of maximizing margin M is identical to the problem of minimizing a function *L(α)* subject to some constraints. To classify all the training examples *zj* correctly, there is a constraint model which is the requirement for the hyperplane. Formally the equation is,

$$\underset{\alpha,\alpha_o}{min} L(\alpha) = \frac{1}{2}\,\alpha^2\,\textit{subject to}\,\mathrm{x_j}\,\alpha^{\mathrm{c}}\,z_{\mathrm{j}}+\alpha_o \geq 1 \tag{12}$$

Where *xj* represents the labels of training examples.

This is actually a lagrangian optimization problem which can be solved by using the lagrange multipliers in order to obtain the weight vector α and the α_o bias of optimal hyperplane.

Propose positive lagrange multipliers, one for each of the inequality constraints. This offers lagrangian:

$$L_{\mathrm{m}} = \frac{1}{2}U^2 - \sum\nolimits_{j=1}^{i} B_{\mathrm{J}}\,\mathrm{X_J}(\mathrm{z_J.u} - \mathrm{c}) + \sum\nolimits_{J=1}^{i} B_{\mathrm{J}} \tag{13}$$

Minimize *Lp* with relevance to u, c. This is a convex quadratic programming problem. In the solution, those points for which $\beta_{\mathrm{J}} > 0$ are called “support vectors”. Model selection of SVM is also a difficult approach. SVM has shown a good performance in data classification recently. Tuning of several parameters is an effective approach which affects the generalization error and this acts as the model selection procedure. In case of linear SVM there is a need to tune the cost parameter C. However, linear SVM is generally applied to linearly separable problems. In cross validation, grid search method can be used to find the paramount parameter set. Then we obtain the classifier after applying this parameter set to the training dataset and this classifier is used to classify the testing dataset to obtain the generalization accuracy [15].

3 Experimental Results

3.1 Dataset

The dataset is collected from the Egyptian Liver Research Institute and the Mansoura Central Hospital, Dakahlia Governorate, Egypt. Till data, there is no availability of standard big dataset that is used for diagnosing of liver diseases in Egypt using conventional factors. Therefore, these databases are collected by efforts of individual research groups. Each collected data contains of the lesions of the liver includes alcoholic liver damage (ALD), primary hepatoma (PH), liver cirrhosis (LC), cholelithiasis (C), and HCC. We have collected 7000 patient’s data, in which the patient’s age in the dataset ranges from 4 to 90 years. Table 1 shows the details of the collecting data. A dataset was developed with twenty three attributes as shown in Table 1 that include the records of 7000 patients in which 5295 patients were male and rests were female.

Table 1. Attributes from model for liver diseases diagnosis

No.	Attribute name (unit)	Range
1	Age (years)	[4–90]
2	General	[Male–Female]
3	BMI Body Mass Index	[18–25]
4	Hemoglobin (g/l)	[123–174 g/L]
5	RBC (10^6/µl)	[0.2–1.6]
6	WBC (10^3/µl)	[4.0–10.0]
7	INR	[1.7–2.3]
8	ALP (IU/l)	[35–100]
9	TB :Total Bilirubin (mg/dl)	[0.4–75]
10	DB: Direct Bilirubin (mg/dl)	[0.1–19.7]
11	GGTP (IU/l)	[3.0–35]
12	Na (mmol/l)	[135 to 145]
13	K (mmol/l)	[3.5–5.5]
14	Cholesterol (mg/dl)	[136–145]
15	TP: Total Protein (g/dl)	[2.7–9.6]
16	ALB: Albumin (g/dl)	[0.9–5.5]
17	A/G Ratio: Albumin and Globulin Ratio (%)	[0.3–2.8]
18	Alkphos: Alkaline Phosphotase	[63–2110]
19	Sgpt Alamine :Aminotransferase	[10–2000]
20	Sgot Aspartate: Aminotransferase	[10–4929]
21	PT prothrombin time(S)	[11–15]
22	AST aspartate aminotransferase(S)	[0–35]
23	ALT alanine aminotransferase(S)	[0–35]

Where RBC red blood cells; WBC white blood cells; PT prothrombin time; INR international normalized ratio; AST aspartate aminotransferase; ALT alanine aminotransferase; ALP alkaline phosphatse; Na natrium; K kalium [16, 17].

3.2 Classifications Performance

Our study focuses on integrating machine learning with individual and hybrid classifiers (NB, C5.0, and SVM). In this proposed technique, the machine learning is employed as a feature selection technique and C5.0, SVM, and NB are employed as an ensemble model. The proposed machine learning showing in Fig. 2.

The dataset is collected from the Egyptian Liver Research Institute and the Mansoura Central Hospital, Dakahlia Governorate, Egypt.

The dataset should be prepared to eliminate the redundancy, and check for the missing values. The data preparation process is often the mainly time-consuming and computational stage. In this step, data sets have been split into the training set, which is used to build the machine learning and testing set that is used to evaluate the proposed machine learning technique.

The dataset is first divided as (90% of them are used for classifier training and 10% for classifier testing). The accuracies from each of the dataset are averaged to given an overall accuracy. It avoids the problem of overlapping test dataset and makes optimal

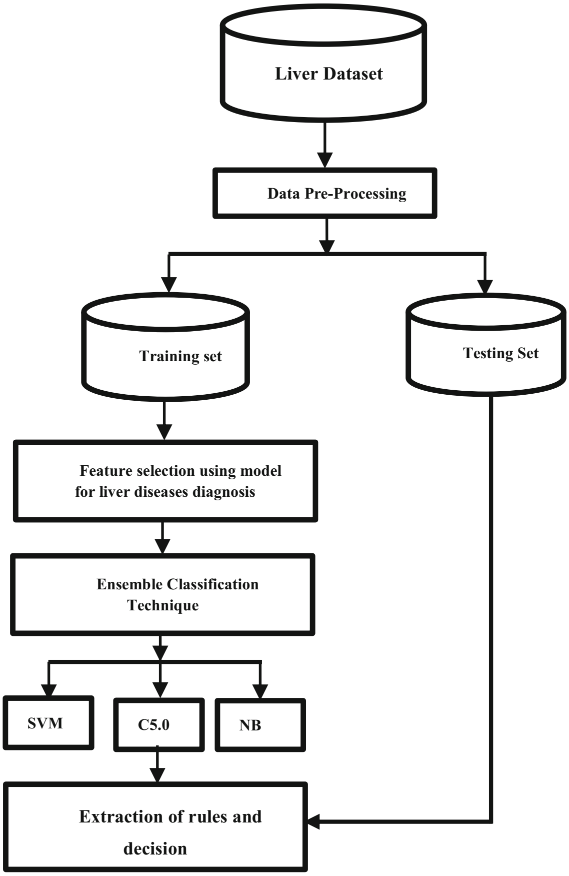

Liver Dataset
Data Pre-Processing
Training set
Testing Set
Feature selection using model for liver diseases diagnosis
Ensemble Classification Technique
SVM
C5.0
NB
Extraction of rules and decision

Fig. 2. The architecture of the proposed techniques.

employment of the obtainable data. We used accuracy, sensitivity, and specificity to test the classification performance of the proposed technique.

We used as a feature selection technique to generate a subset of features from the original features that make machine learning easier and less time-consuming. After generating a subset, NB, C5.0, and SVM are separately used as classification techniques. Then, an ensemble classifier (NB + C5.0 + SVM) is proposed to classify the data.

- Accuracy is the percent of correct classifications and can be defined as:

$$Accuracy = \frac{\mathrm{TP} + \mathrm{TN}}{TP + FP + TN + FN} \times 100 \tag{14}$$

- Sensitivity is the rate of true positive and can be defined as:

$$= \frac{TP}{TP + FN} \times 100 \tag{15}$$

- Specificity is the true negative rate and can be defined as:

$$= \frac{TN}{TN + FP} \times 100 \tag{16}$$

Where:

TP = the number of positive examples correctly classified.
FP = the number of positive examples misclassified as negative
FN = the number of negative examples misclassified as positive
TN = the number of negative examples correctly classified [5].

Table 2 shows the accuracy, sensitivity, and specificity of our proposed machine learning before and after the fine-tuning. From the table, it can be noticed that the accuracy of the technique after fine-tuning the overall is higher than the accuracy before the fine-tuning. Thus, the fine-tuning step is very important for improving the accuracy of classification.

Table 2. Performance of the proposed machine learning before and after

Measurements	Accuracy (%)	Sensitivity (%)	Specificity (%)
Before proposed	90.50	91.60	88.50
After proposed	97.20	98	95.70

Figure 3 shows a diagram that represents the overall performance evaluation of the proposed technique SVM, C5.0, and Naïve Bayes. It is shown that the accuracy, sensitivity, and specificity of our proposed technique are better than the three state-of-the-art techniques.

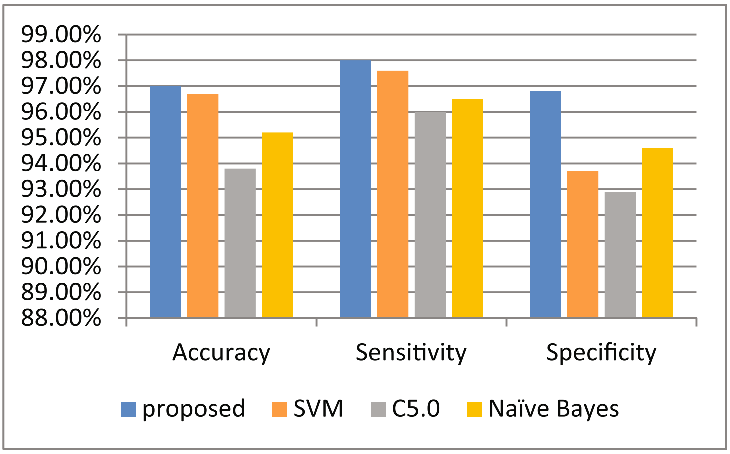


Fig. 3. The overall comparison between the proposed with SVM, C5.0, and Naïve Bayes.

Figure 4 shows ten relevant factors in the prediction of liver disease according to our technique. The rules generated by our technique is shown in Table 3. According to the Table 1, it can be seen that 9 rules have been produced by our technique.

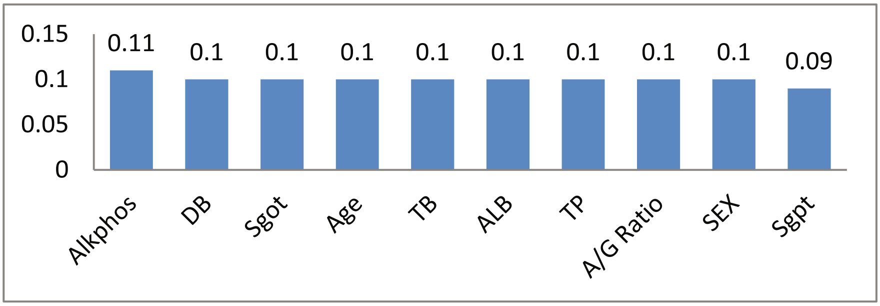


Fig. 4. The importance of the factors in the prediction of liver disease by using our technique.

Table 3. Some rules generated by our technique

No.	Rules
1	IF DB <= 1 and Age > 17 and Sgot <= 1.30 and Age <= 65 and Age > 58 and A/G > 1.390 THEN class 1.0
2	IF DB <= 3.600 and Sgot <= 64 and DB <= 0.300 and Alkphos > 130 and DB > 0.100 and TB <= 0.700 and ALB <= 2.300 THEN class 2.0
3	IF DB <= 3.600 and Sgot <= 64 and DB <= 0.300 and Alkphos > 130 and DB > 0.100 and TB <= 0.700 and ALB > 2.300 and TB > 0.600 and ALB <= 4.200 and Sgpt <= 36 and Sgot <= 14 THEN class 1.0
4	IF DB <= 1.200 and Sgpt <= 65 and TB <= 1.600 and Alkphos <= 211 and DB <= 0.100 and SEX = Female Alkphos > 153 and TB <= 0.600 and A/G <= 0.950 THEN class 2.0
5	IF DB <= 3.600 and Sgot <= 64 and DB <= 0.300 and Alkphos > 130 and DB > 0.100 and TB <= 0.700 and ALB > 2.300 and TB <= 0.600 THEN class 1.0
6	IF DB <= 3.600 and Sgot <= 64 and DB > 0.300 and TB <= 5.900 and Age > 38 and DB <= 1 and TP <= 6.200 and Alkphos > 314 THEN class 1.0
7	IF DB <= 3.600 and Sgot <= 64 and DB > 0.300 and TB <= 5.900 and Age > 38 and DB <= 1 and TP <= 6.200 and Alkphos <= 314 THEN class 2.0
8	IF DB <= 3.600 and Sgot > 64 and ALB > 2.200 and Sgot <= 298 and TP > 5.200 and Age <= 39 and TP <= 7.900 THEN class 1.0
9	IF DB <= 1.200 and Sgpt <= 65 and TB <= 1.600 and Alkphos <= 211 and DB <= 0.100 and SEX = Female and Alkphos > 153 and TB > 0.600 THEN class 1.0

4 Conclusion and Feature Work

In this paper, we proposed and built a machine learning based on a hybrid classifier to be used as a classification model for liver diseases diagnosis to improve performance and experts to identify the chances of disease and conscious prescription of further treatment healthcare and examinations.

In future work, the use of fast datasets technique like Apache Hadoop or Spark can be incorporated with this technique. In addition to this, we can use distributed refined algorithms like Forest Tree implemented in Apache Hadoop to increase scalability and efficiency.

References

1. http://www.who.int/countries/egy/en/
2. Kumar, Y., Sahoo, G.: Prediction of different types of liver diseases using rule based classification model. Technol. Health Care **21**(5), 417–432 (2013)
3. Roy, S., Singh, A., Shadev, S.K.: Machine learning method for classification of liver disorders. Far East J. Electron. Commun. **16**(4), 789 (2016)
4. Zhang, Y., et al.: A systems biology-based classifier for hepatocellular carcinoma diagnosis. PLoS One **6**(7), e22426 (2011)
5. Kavakiotis, I., et al.: Machine learning and data mining methods in diabetes research. Comput. Struct. Biotech. J. **15**, 104–116 (2017)
6. Ayodele, T.O.: Types of machine learning algorithms. In: New Advances in Machine Learning. InTech (2010)
7. Gullo, F.: From patterns in data to knowledge discovery: what data mining can do. Phys. Procedia **62**, 18–22 (2015)
8. Pandey, B., Singh, A.: Intelligent techniques and applications in liver disorders. Survey, January 2014
9. Takkar, S., Singh, A., Pandey, B.: Application of machine learning algorithms to a well defined clinical problem: liver disease. Int. J. E-Health Med. Commun. (IJEHMC) **8**(4), 38–60 (2017)
10. Siuly, S., Zhang, Y.: Medical big data: neurological diseases diagnosis through medical data analysis. Data Sci. Eng. **1**(2), 54–64 (2016)
11. Luo, J., et al.: Big data application in biomedical research and health care: a literature review. Biomed. Inform. Insights **8**, 1 (2016)
12. Ramana, B.V., Babu, M.S.P., Venkateswarlu, N.B.: A critical study of selected classification algorithms for liver disease diagnosis. Int. J. Database Manag. Syst. (IJDMS) **3**(2), 1–14 (2011)
13. Bujlow, T., Riaz, T., Pedersen, J.M.: A method for classification of network traffic based on C5.0 machine learning algorithm. In: 2012 International Conference on Computing, Networking and Communications (ICNC). IEEE (2012)
14. Ozcift, A., Gulten, A.: Classifier ensemble construction with rotation forest to improve medical diagnosis performance of machine learning algorithms. Comput. Methods Programs Biomed. **104**(3), 443–451 (2011)

15. Fatima, M., Pasha, M.: Survey of machine learning algorithms for disease diagnostic. J. Intell. Learn. Syst. Appl. **9**(01), 1 (2017)
16. Singh, A., Pandey, B.: Intelligent techniques and applications in liver disorders: a survey. Int. J. Biomed. Eng. Technol. **16**(1), 27–70 (2014)
17. http://www.scymed.com/en/smnxpc/pcdcc770.htm

Stance Detection in Tweets Using a Majority Vote Classifier

Sara S. Mourad[1(✉)], Doaa M. Shawky[1,2], Hatem A. Fayed[1,2], and Ashraf H. Badawi[2]

[1] Engineering Mathematics Department, Faculty of Engineering, Cairo University, Giza, Egypt
saracherif92@gmail.com, doaashawky@staff.cu.edu.eg
[2] Zewail City of Science and Technology, Giza, Egypt
{hfayed,abadawi}@zewailcity.edu.eg

Abstract. The task of stance detection is to determine whether someone is in favor or against a certain topic. A person may express the same stance towards a topic using positive or negative words. In this paper, several features and classifiers are explored to find out the combination that yields the best performance for stance detection. Due to the large number of features, ReliefF feature selection method was used to reduce the large dimensional feature space and improve the generalization capabilities. Experimental analyses were performed on five datasets, and the obtained results revealed that a majority vote classifier of the three classifiers: Random Forest, linear SVM and Gaussian Naïve Bayes classifiers can be adopted for stance detection task.

Keywords: Stance detection · Random Forest · Support vector machine
Gaussian Naïve Bayes · ReliefF · Tweets

1 Introduction

In the era of big data, it is complex for humans to analyze debates towards a topic. The need for an automatic stance detection model appears in many applications such as political campaigns.

The goal of this paper is to develop a model for an automatic detection of the stance described by the tweets that have been released by the international workshop on Semantic Evaluation (SemEval) [1]. The release of the first dataset of tweets annotated for both stance and sentiment by the SemEval workshop, organized in 2016, helps to address the interaction between the detection of sentiment and stance [1].

In this paper, we address one of the SemEval tasks, the stance detection task, which includes two subtasks. The first one, subtask A, is the supervised task. It contains a dataset of about 4000 tweets, each tweet expresses a specific stance towards one of the five targets: 'Atheism', 'Climate Change is a Real Concern', 'Feminist Movement', 'Hillary Clinton', and 'Legalization of Abortion'. Tweets are annotated also whether the target of opinion in the tweet is the target we are interested in. On the other hand, the second subtask; subtask B, is the unsupervised task to detect the stance towards the target "Donald Trump". The dataset available for "Trump" contains about 78000 unlabeled tweets [1].

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 375–384, 2018.
https://doi.org/10.1007/978-3-319-74690-6_37

Sentiment analysis (SA) and stance detection (SD) tasks are related but different; SD is more similar to target-dependent sentiment classification [2].

SA models detect whether the text is negative, positive, or neutral, whereas SD models detect favorability of a topic of interest and label the text as “Favor”, “Against” or “None”. “None” label in SD indicates that the text is not related to the target or it is neutral towards the target [1].

A tweet may be in favor of a topic but conveys negative sentiment, consider for example the following tweet that is in favor of the target “Atheism”: “*Impressed with the nice Tunisians. Not impressed with religion; a major cause of so much heartache & death. #remembertunisia*”

The difficulty of the SD task stems from that the target of opinion may not be expressed explicitly in the text, or the target of opinion may not be the same as the target of interest [1].

For example, the following tweet implies favorability to the target of interest “Hillary” by opposing its opponent “Trump”: “*The moment you opened your mouth that was the end of your candidate trip! @realDonaldTrump*”.

Moreover, the following tweet does not mention explicitly the target of opinion, which is “Hillary Clinton”: *“@donnabrazile @DougHeye So this woman does one thing to help. She’s GOP - she’s, ultimately, the enemy - don’t doubt it*”.

The rest of the paper is organized as follows. Section 2 introduces the previous work that was performed on the SemEval dataset for stance detection. Section 3 includes the description of the different stages of the proposed model. In addition, Sect. 4 describes the experimental study. The obtained results are discussed in Sect. 5. Finally, Sect. 6 draws the conclusions and highlights the direction for the future work.

2 Related Work

Previous work on stance detection using SemEval dataset was provided by the research teams that participated in the SemEval workshop. Some works [3–8] were published after the workshop and outperformed the top ranked teams [9, 10]. In the following, the top-ranked approaches introduced in the SemEval workshop are presented, and then the approaches that outperformed the top teams are highlighted.

A recurrent neural network (RNN) formed from long short-term memory (LSTM) units [11] was used in [9] to address Task A. The authors used additional tweets in the generation of the embeddings using word2vec [12] and during the training of RNN.

In addition, a convolutional neural network (CNN) architecture following [13] was used in [10] to address both tasks. The main contribution of this work was in designing a vote scheme for the prediction. For the unsupervised task, the authors built a two-class dataset and used softmax to classify the data into the three classes (Favor, Against, or None).

Moreover, it was proved in [3] that bidirectional conditional LSTM encoding outperforms independent encoding. In bidirectional conditional encoding, both right-hand side and left-hand side contexts are used to get a target-dependent representation of the tweet. The input to LSTM cells were word vectors obtained by word2vec trained on all the available tweets and additional collected tweets. The authors automatically labeled the unlabeled dataset provided by the workshop for "Trump". This paper addressed only Task B.

Some of the SemEval organizers built a stance detection model [4]. Their approach outperformed the participants' ones. The authors addressed Task A using linear SVM trained on word and character grams. In addition, they used a binary feature indicating the presence of the target in the tweet, and word embedding features generated using large collection of tweets.

The work in [5] proved that it is crucial to include context-based features to improve stance detection models. The authors were interested in political tweets only, so only "Hillary" dataset was used in Task A. For both tasks, "Hillary" dataset was used for training a Gaussian Naïve Bayes classifier. Features used include: structural features (frequency of hashtags and mentions), labeled-based features, context-based features, and sentiment-based features using lexicons. For labeled-based features, the sentiment and opinion manual labels released by the workshop were used. Context-based features were obtained using lists of words related to the target.

The unsupervised task was the focus of the work presented in [6]. The authors labeled "Trump" dataset for stance using hinge-loss Markov random fields, and then used probabilistic soft logic to augment the weakly-labeled tweets by incorporating relational features such as user profile and friendship information. The augmented dataset was used as training set for linear SVM. Task A labeled datasets were used as the development set. Features used include: word grams, character grams, and sentiment features.

Also, a linear SVM trained on word and character grams was used in [7] for both tasks. The authors presented a log-linear model in which stance and sentiment variables were used in a novel way of multi-way interactions. VADER [14], the sentiment analyzer for tweets, and LDA [15] were used to get the sentiment labels. Their model generalized across multiple targets. For the unsupervised task, the authors collected additional tweets and automatically labeled the tweets based on manually labeled hashtags.

Recently, a neural attention model that combines LSTM and target-specific attention extractor was proposed in [8] to address Task A. The proposed approach outperformed top performed models that used SemEval English dataset in addition to those that used Chinese dataset for stance detection.

3 The Proposed Model

This paper addresses only Task A, the supervised task, from SemEval-2016 workshop. The proposed model consists of three stages: preprocessing, feature extraction/selection and classifier design as shown in Fig. 1. Each stage is described in details below.

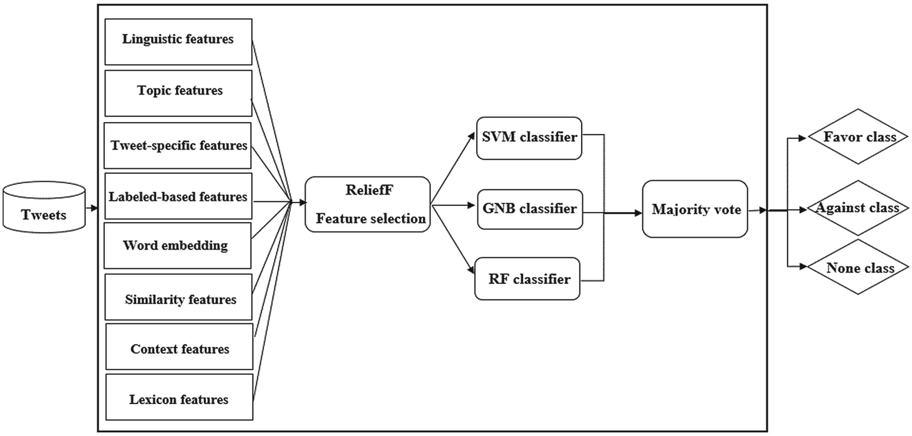


Fig. 1. Flow diagram of the model.

3.1 Preprocessing

The hashtag "#SemST" and stop words were removed from the tweets. However, negation words were not removed. In addition, concatenated words in hashtags and mentions were separated.

3.2 Feature Extraction/Selection

In this stage, several features were extracted to represent the tweets. The description of each feature is described below. Due to the large number of features (about 43000 columns), we used ReliefF [16], the feature selection algorithm, to reduce the number of features. ReliefF method assigns a weight to each feature, reflecting its importance; the weight of the feature increases if it differs from that feature in nearby samples of the other class more than nearby instances of the same class.

Linguistic features

The used linguistic features can be classified into word, character, part-of-speech (POS), dependency and cluster features.

Word features include word 1–3 grams computed as: binary vectors, count-based vectors, and tf-idf weighted vectors with a frequency cut-off of two.

Character features include Character 1–6 grams computed as count-based vectors.

Part-of-speech (POS) features

- 1–3 grams for the tweet POS identified using CMU Twitter NLP tool [17].
- 1–3 grams obtained after concatenating each word in the tweet with its POS generated by NLTK library [18], i.e. the tweet was transformed to: word0_pos0 word1_pos1 word2_pos2.

- 1–3 grams obtained after concatenating the whole tweet and POS corresponding to words found in the tweet, i.e. the tweet was transformed to: word0 word1 word2 pos0 pos1 pos2.

For dependency features, dependency relations were extracted using Stanford parser [19]. Stanford dependencies provides a representation of grammatical relations between words in a sentence. For each tweet, several triplets of name of the relation (rel), governor (gov), and dependent (dep) were generated. We used a binary feature for each of the three tuples (rel, gov, dep), (rel, gov), and (gov, dep) found in the corpus.

As for the cluster features, we computed, for each tweet, the number of words belonging to each of the 1000 clusters generated using the Brown Algorithm provided by the CMU TweetParser tool [17].

Topic features

Topic features were extracted using LDA [15]. These features were used in [20]. Topic features include the following.

Sentence to topic features: the probability that the tweet belongs to each topic of the different topics was used.

Word to topic features: the accumulation of the probability of each topic over all words of the tweet was computed.

Top topic words features: for each tweet, we counted the occurrence of each of the top topic words found in the tweet, the total number of top words found in the tweet per topic, and the ratio of the number of top words found to the total number of top words for each topic.

Tweet-specific feature

Tweet-specific features include tweet length, average word length, hashtag features, mention features, conditional sentences, and counting features.

Hashtag features: Hashtags 1–3 grams were computed as count-based vectors, binary vectors and tf-idf weighted vectors. In addition, hashtags 1–3 grams were extracted after replacing target words by a specific flag. (For e.g. For “Hillary” dataset, target words are: Hillary, Clinton …)

Mention features: Mentions 1–2 grams were computed using count-based vectors and binary vectors. In addition, mentions 1–2 grams were extracted after replacing target words by a specific flag.

Counting features: The number of negation, elongated, misspelled, capitalized words, hash and mention symbols, and punctuation marks.

Labeled-based features

We used the sentiment and opinion labels provided by the SemEval workshop after evaluating the participant teams.

Sentiment label for a tweet can be “positive”, “negative”, or “other”.

Opinion label can be one of the three following labels: “1. The tweet explicitly expresses opinion about the target, a part of the target, or an aspect of the target.”, “2. The tweet does NOT expresses opinion about the target but it HAS opinion about something or someone other than the target.", " 3. The tweet is not explicitly expressing opinion. (For example, the tweet is simply giving information)".

Word embedding features

The word vectors generated by skip-gram model from word2vec tool [12] trained over the tweets were used as features. The vector of the tweet is obtained by averaging the vectors of the words. In addition, we weighted the vector of each word by information count. The information count of a word is the ratio of the frequency of all words to the frequency of this word in the corpus.

Similarity features

The cosine similarity was computed by applying a dot product between the word vector of the tweet and the vector of the target word.

Context features

Context-based features are used to capture information related to the domain of the target. For each target, we manually constituted three lists: a list containing central words related to target, another list for words indicating favorability, and the third list for words indicating opposition. These lists were created after manually investigating the datasets.

Context features for "Hillary" dataset were generated following [5] considering different relationships between the target and the entities related to it. There are five lists manually created to check if:

- The target is mentioned by name explicitly in the tweet (Hillary/Hilary/cliton/Clinton/hill).
- The target is referenced implicitly by pronoun (She/her).
- The party of the target is mentioned (democratic/democrat/…).
- One of the party colleagues of the target is mentioned (Bernie/Sanders/…).
- One of the opponents that belong to the rival party of the target is mentioned (Ted/Cruz/Marco/…).

In addition, we automatically created for all targets two lists containing keywords that tend to indicate favorability or opposition to the target. The keywords were chosen based on the frequent words in favor and against tweets.

For each tweet, we computed the count of the words belonging to each of the previously mentioned lists. In addition, we used binary features indicating the presence of any words belonging to these lists. The process was repeated for the hashtags found in each tweet, we used features indicating whether these hashtags contain words belonging to these lists.

Sentiment lexicon features

General-inquirer lexicon [21]: this lexicon contains about 80 categories, each word in the lexicon may belong to many categories. We used only 6 categories: positive, negative, hostile, strong, pleasure, and pain. For each tweet, we counted the words belonging to each of the six categories.

AFINN lexicon [22]: AFINN contains English words with polarity between −5 and +5. For each tweet, we computed the sum of scores for words found in the lexicon.

DAL lexicon [23]: Each word in the lexicon has a score for pleasantness, imagery, and activation. For each tweet, the sum and the average of the pleasantness, imagery, and activation scores of words were computed.

MPQA [24], *Bing Liu lexicons* [25], *and SentiWordNet* [26]: for each tweet, we computed the ratio of number of each of the positive and negative words to the total number of words. Also, we computed the sum of each of the positive and negative words in the tweet. In addition, some features were generated following [27]; we counted the number of words in positive/negative contexts belonging to the positive/negative lexicons. The process was repeated for the hashtags. For each tag of the POS tags, we counted the number of words in positive/negative contexts that belong to the positive/negative lexicon. These features are generated using each of the three lexicons.

NRC hashtag and NRC sentiment 140 lexicons [28, 29]: for each tweet, we computed the sum of the scores of the words, max score, min score, number of positive words, and number of negative words.

3.3 Classifier Design

In this stage, three classifiers were established, namely, Random Forest (RF), Gaussian Naïve Bayes (GNB), and linear SVM. Then the majority vote classifier was employed to obtain the final class.

4 Experimental Study

To choose the classier, we started by applying all features to "Hillary" dataset and using three classifiers from Scikit-learn library [30]: RF, GNB, and linear SVM. Using 5-fold cross-validation, we tuned the parameters of each classifier.

We tuned the parameters of ReliefF by varying them in a range of values and following the direction of the increase of the F-score. First, the number of neighbors' features to be considered was fixed at its default value, while the number of features to be kept was varied. The performance of fifty features was the best as shown in Table 1. Second, the number of features was fixed at fifty, while the number of neighbors was varied as shown in Table 2. Finally, the number of neighbors was set to 150. Using ReliefF, the most significant features are the labeled-based features, word grams, character grams, POS features, and hashtag features.

Table 1. Tuning the number of features in ReliefF. (F-scores are shown)

Neighbors	Features	GNB	RF	SVM
100	30	0.5360	0.7603	**0.7508**
100	50	**0.5433**	**0.7754**	0.7379
100	70	0.5253	0.7409	0.7390
100	100	0.4252	0.7618	0.6868

Table 2. Tuning the number of neighbors in ReliefF. (F-scores are shown)

Neighbors	Features	GNB	RF	SVM
50	50	**0.5496**	0.7647	0.7143
100	50	0.5433	0.7754	0.7379
150	50	0.5353	**0.7857**	**0.7442**
200	50	0.5263	0.7776	0.72955

5 Results and Analysis

It was found that RF performs well for the targets "Feminist" and "Hillary" with F-scores of 0.46 and 0.785, respectively. On the other hand, SVM performs better for "Atheism" and "Abortion" datasets, with F-scores of 0.7 and 0.64 respectively. Whereas, an F-score of 0.55 is obtained for "Climate" dataset using GNB. Thus, we built a majority vote classifier formed from the three classifiers.

The evaluation metric is the macro-averaged F-score of the *Favor* and *Against* classes. *None* class is not disregarded: by taking the average of *Favor* and *Against* F-scores, *None* class is treated as a class that is not of interest. The performance of the proposed model is shown in Table 3. To evaluate our model, we listed the performance of the top-ranked teams Mitre [9] and Pkudblab [10]. In addition, the table shows Task A scores obtained by the papers published after the workshop: Joint Model [7], organizers model [4], and Target-Specific Neural Attention Networks (TAN) [8].

Table 3. Comparison of F-score of different models for SemEval Task A

Dataset	Joint model	Organizers model	Our model	TAN	Mitre	Pkudblab
Hillary	0.7448	0.578	**0.7987**	0.6538	0.5767	0.6441
Abortion	**0.6994**	0.669	0.6272	0.6372	0.5728	0.6109
Atheism	0.6709	0.683	**0.7211**	0.5933	0.6147	0.6334
Climate	0.5004	0.438	0.4680	**0.5359**	0.4163	0.5269
Feminist	0.5790	0.584	0.4256	0.5577	**0.6209**	0.5133
Overall	**0.7103**	0.703	0.7004	0.6879	0.6782	0.6733

As shown in Table 3, the proposed model outperformed all the other models for "Hillary" and "Atheism" datasets. In addition, a competitive overall score was obtained, which indicates that the proposed model can be efficiently adopted for stance detection. Our model is feature-based model: it does not rely on neural networks (unlike Pkudblab, Mitre, and TAN models) or additional data (unlike Mitre and organizers models).

6 Conclusions and Future Work

In this paper, a new model was proposed for stance detection. In this model, several features were extracted and combined to develop a majority vote classifier among RF, GNB, SVM classifiers since each classifier performed well for specific datasets.

Promising results were obtained across all the five datasets studied. In the future work, we will include irony detection. In addition, methods to automatically generate sentiment and opinion labels with an accuracy near to that of the manually labels provided by the workshop will be explored.

References

1. Mohammad, S., Kiritchenko, S., Sobhani, P., Zhu, X., Cherry, C.: SemEval-2016 Task 6: detecting stance in tweets. In: Proceedings of SemEval, pp. 31–41 (2016)
2. Jiang, L., Yu, M., Zhou, M., Liu, X., Zhao, T.: Target-dependent Twitter sentiment classification. In: Proceedings of ACL, pp. 151–160 (2011)
3. Augenstein, I., Rocktäschel, T., Vlachos, A., Bontcheva, K.: Stance detection with bidirectional conditional encoding. In: Proceedings of EMNLP, pp. 876–885 (2016)
4. Mohammad, S., Sobhani, P., Kiritchenko, S.: Stance and sentiment in tweets. Spec. Sect. ACM Trans. Internet Technol. Argum. Soc. Media , **17**(3), 26 (2017)
5. Lai, M., Farías, D.I.H., Patti, V., Rosso, P.: Friends and enemies of Clinton and Trump: using context for detecting stance in political tweets. In: Mexican International Conference on Artificial Intelligence, pp. 155–168 (2016)
6. Ebrahimi, J., Dou, D., Lowd, D.: Weakly supervised tweet stance classification by relational bootstrapping. In: Proceedings of EMNLP, pp. 1012–1017 (2016)
7. Ebrahimi, J., Dou, D., Lowd, D.: A joint sentiment-target-stance model for stance classification in tweets. In: Proceedings of COLING, pp. 2656–2665 (2016)
8. Du, J., Xu, R., He, Y., Gui, L.: Stance classification with target-specific neural attention networks. In: Proceedings of IJCAI, pp. 3988–3994 (2017)
9. Zarrella, G., Marsh, A.: MITRE at SemEval-2016 Task 6: transfer learning for stance detection. In: Proceedings of SemEval, pp. 458–463 (2016)
10. Wei, W., Zhang, X., Liu, X., Chen, W., Wang, T.: Pkudblab at SemEval-2016 Task 6: a specific convolutional neural network system for effective stance detection. In: Proceedings of SemEval, pp. 384–388 (2016)
11. Li, J., Luong, T., Jurafsky, D.: A hierarchical neural autoencoder for paragraphs and documents. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, vol. 1, pp. 1106–1115 (2015)
12. Mikolov, T., Chen, K., Corrado, G., Dean, G.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
13. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1746–1751 (2014)
14. Hutto, C.J., Gilbert, E.: Vader: a parsimonious rule-based model for sentiment analysis of social media text. In: Eighth International Conference on Weblogs and Social Media (ICWSM), pp. 216–255 (2014)
15. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
16. Kononenko, I., Šimec, E., Robnik-Šikonja, M.: Overcoming the myopia of inductive learning algorithms with RELIEFF. J. Appl. Intell. **7**(1), 39–55 (1997)
17. Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N., Smith, N.A.: Improved part-of-speech tagging for online conversational text with word clusters. In: HLT-NAACL, pp. 380–390 (2013)

18. Loper, E., Bird, S.: NLTK: the natural language toolkit. In: Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics, pp. 63–70. ACL (2002)
19. Klein, D., Manning, C.D.: Accurate unlexicalized parsing. In: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics, vol. 1, pp. 423–430 (2003)
20. Zhang, Z., Lan, M.: ECNU at SemEval 2016 Task 6: relevant or not? Supportive or Not? A two-step learning system for automatic detecting stance in tweets. In: Proceedings of SemEval, pp. 451–457 (2016)
21. Stone, P., Dumphy, D., Smith, M., Ogilvie, D.: The General Inquirer: A Computer Approach to Content Analysis. MIT Studies in Comparative Politics. MIT Press, Cambridge (1966)
22. Nielsen, F.Å.: A new ANEW: evaluation of a word list for sentiment analysis in microblogs. arXiv preprint arXiv:1103.2903 (2011)
23. Whissell, C.: Using the revised dictionary of affect in language to quantify the emotional undertones of samples of natural language. Psychol. Rep. **105**(2), 509–521 (2009)
24. Wilson, T., Wiebe, J., Hoffmann, P.: Recognizing contextual polarity in phrase-level sentiment analysis. In: Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, pp. 347–354. Association for Computational Linguistics (2005)
25. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 168–177. ACM (2004)
26. Baccianella, S., Esuli, A., Sebastiani, F.: Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In: Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC), Valletta, Malta, pp. 2200–2204. European Language Resources Association (ELRA) (2010)
27. Balikas, G., Amini, M.R.: TwiSE at SemEval-2016 Task 4: Twitter sentiment classification. In: Proceedings of SemEval, pp. 85–91 (2016)
28. Kiritchenko, S., Zhu, X., Mohammad, S.: Sentiment analysis of short informal texts. J. Artif. Intell. Res. **50**, 723–762 (2014)
29. Mohammad, S., Kiritchenko, S., Zhu, X.: NRC-Canada: building the state-of-the-art in sentiment analysis of tweets. In: Proceedings of SemEval, pp. 321–327 (2013)
30. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)

A Comparative Study of Classification Methods for Flash Memory Error Rate Prediction

Barry Fitzgerald[1](✉), Jeannie Fitzgerald[2], Conor Ryan[3], and Joe Sullivan[1]

[1] Limerick Institute of Technology, Limerick, Ireland
barry.fitzgerald@lit.ie
[2] NVMdurance Ltd., Limerick, Ireland
[3] University of Limerick, Limerick, Ireland

Abstract. NAND Flash memory has been the fastest growing technology in the history of semiconductors and is now almost ubiquitous in the world of data storage. However, NAND devices are not error-free and the raw bit error rate (RBER) increases as devices are programmed and erase (P-E cycled). This requires the use of error correction codes (ECCs), which operate on chunks of data called codewords. NAND manufacturers specify the number of P-E cycles a device can tolerate (known as endurance) very conservatively to account for quality variations within and across devices. This research uses machine learning to predict the true cycling level each part of a NAND device can tolerate, based on measurements taken from the device as it is used. Real data is gathered on millions of codewords and eight machine learning classification methods are compared. A new subsampling method based on the error probability density function is also proposed.

Keywords: Flash memory · Machine learning · Error rate prediction
Classification · Subsampling

1 Introduction

NAND flash memory has seen an explosion in growth over the last twenty years as the world's insatiable hunger for data storage has grown exponentially. NAND flash is the dominant form of data storage for consumer devices such as mobile phones, digitial cameras, MP3 players, USB drives. As technology scaling drives the cost per bit of NAND flash down, NAND flash-based solid state drives (SSDs) are replacing traditional hard disk drives (HDDs) in home computing [1] and enterprise storage applications [2]. SSDs offer a number of advantages over HDDs, such as faster performance and lower power consumption. However, one of the main disadvantages of flash memory is its limited working lifetime due to wearout through use [3].

NAND devices are divided into regions called *blocks*, which consist of rows of flash cells called *pages*. Devices are programmed on a page basis and erased on a block basis. Each page is subdivided into a number of *sectors* (typically 8 or 16 sectors per page). NAND flash devices are not error-free and require the use of ECCs to recover data. When data is programmed to a sector, ECCs assign

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 385–394, 2018.
https://doi.org/10.1007/978-3-319-74690-6_38

parity bits to the data to form a *codeword*, and use this parity information to detect and correct errors when the codeword is read back.

Flash devices degrade as they are P-E cycled, resulting in an increase in RBER. Eventually the number of errors in a codeword exceeds the level correctable by ECC, resulting in uncorrectable errors [4].

The *endurance* specification of a flash device refers to the number of P-E cycles a device can withstand before the onset of uncorrectable errors. NAND manufacturers specify the endurance rating so that devices are not P-E cycled beyond the point where uncorrectable errors are expected to occur.

This research gathered real-world RBER data on more than 27 million codewords across multiple devices from a current generation NAND flash family. Devices were P-E cycled to multiple cycling levels beyond the endurance specification. We found that no codewords fail (to a typical ECC level) at the rated endurance of 5 K P-E cycles, and that even at 3X the rated endurance the majority of codewords still pass.

This indicates that flash manufacturers specify their endurance ratings extremely conservatively to account for worst-case sectors across a population of devices. This is despite the fact that most devices, and most sectors within a device, will perform much better than this. If a flash end user (such as an SSD manufacturer) could predict how far each sector could be cycled, this would enable them to optimise their NAND management. To the best of our knowledge, we are the first to present such a prediction model.

We used eight machine learning methods to build classification models to predict whether a sector would pass or fail at a given cycling level. Inputs to the model include metrics recorded during testing such as bit error rates and the timing of program and erase operations at various cycling points. These metrics could be obtained from the flash while it is cycling live on an SSD. Results for each model are compared using well known machine learning metrics such as sensitivity, specificity, accuracy and ROC curve.

Because of the imbalanced nature of the dataset (more than 99.97% of codewords passed across all cycling levels), the classifier's majority class needed to be subsampled for each model. We propose a new sampling method, called PDF-based sampling, and compare results against randomly sampled data.

2 Related Research

Previous research has applied machine learning to NAND flash for prediction purposes. Hogan et al. [5] developed a model using Genetic Programming, which P-E cycled NAND blocks to destruction (the point at which program or erase operations could no longer be performed), and used program and erase time to predict the cycling level at which this occurred. That study found that better results could be achieved when program and erase times were measured at more than one P-E cycle point, indicating that the rate of change of these timings is a valuable predictive metric.

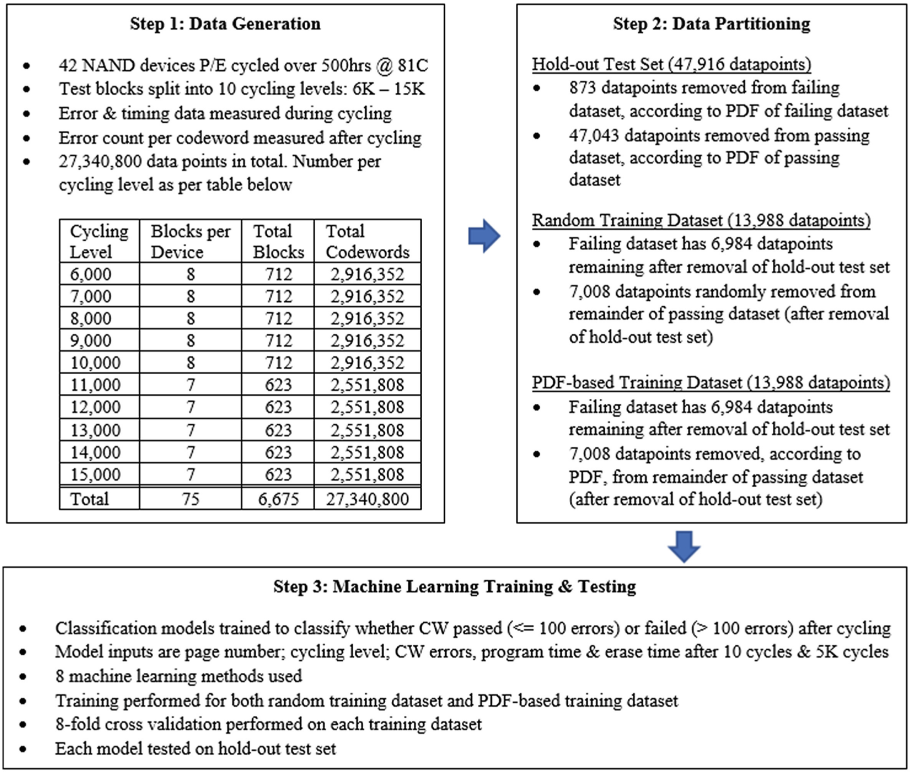
Arbuckle et al. [6] extended this work by comparing different machine learning techniques for creating the prediction model. The classification techniques used were Logistic Regression, Regularised Regression, Naive Bayes, K-Nearest Neighbours, Support Vector Machines and Genetic Programming. Support Vector Machines were found to give the most accurate results.

Our research is fundamentally different to these two studies. Rather than using number of cycles as the classifier output and predicting the point of destruction of blocks, which has limited practical value, we use the number of codeword-level errors to predict the point at which uncorrectable errors occur. This is far more valuable from a practical point of view, as it determines when the true end-of-life occurs.

As well as using program and erase time as inputs to the model, we have identified other metrics, such as errors per codeword and page number, that contribute to the predictive power of the model.

3 Process Overview

An overview of the entire process is given in Fig. 1.



Step 1: Data Generation

- 42 NAND devices P/E cycled over 500hrs @ 81C
- Test blocks split into 10 cycling levels: 6K – 15K
- Error & timing data measured during cycling
- Error count per codeword measured after cycling
- 27,340,800 data points in total. Number per cycling level as per table below

Cycling Level	Blocks per Device	Total Blocks	Total Codewords
6,000	8	712	2,916,352
7,000	8	712	2,916,352
8,000	8	712	2,916,352
9,000	8	712	2,916,352
10,000	8	712	2,916,352
11,000	7	623	2,551,808
12,000	7	623	2,551,808
13,000	7	623	2,551,808
14,000	7	623	2,551,808
15,000	7	623	2,551,808
Total	75	6,675	27,340,800

Step 2: Data Partitioning

Hold-out Test Set (47,916 datapoints)

- 873 datapoints removed from failing dataset, according to PDF of failing dataset
- 47,043 datapoints removed from passing dataset, according to PDF of passing dataset

Random Training Dataset (13,988 datapoints)

- Failing dataset has 6,984 datapoints remaining after removal of hold-out test set
- 7,008 datapoints randomly removed from remainder of passing dataset (after removal of hold-out test set)

PDF-based Training Dataset (13,988 datapoints)

- Failing dataset has 6,984 datapoints remaining after removal of hold-out test set
- 7,008 datapoints removed, according to PDF, from remainder of passing dataset (after removal of hold-out test set)

Step 3: Machine Learning Training & Testing

- Classification models trained to classify whether CW passed (<= 100 errors) or failed (> 100 errors) after cycling
- Model inputs are page number; cycling level; CW errors, program time & erase time after 10 cycles & 5K cycles
- 8 machine learning methods used
- Training performed for both random training dataset and PDF-based training dataset
- 8-fold cross validation performed on each training dataset
- Each model tested on hold-out test set

Fig. 1. Process flowchart.

Each stage of the process is discussed in detail in the following sections.

4 Experimental Setup

4.1 Data Collection

A test system was developed, capable of P-E cycling NAND devices and measuring the number of bit errors per codeword at various cycling points. The test system also had the ability to measure the timing of program and erase operations.

P-E cycling was performed over 500 h at 81° C, as per the industry standard for testing the endurance of SSDs [7]. When NAND devices are P-E cycled on an SSD, each cycle causes damage to the flash cells. This damage recovers between cycles, and the rate of recovery increases with increasing temperature [8]. This means that, for endurance testing, cycling can be performed at a much faster rate than would occur in the field, by performing the cycling at an elevated temperature.

In total, 6,675 blocks were tested across 45 devices. Blocks were split into ten cycling levels: from 6 K to 15 K cycles in 1 K steps. For each device, eight blocks were assigned to each of the 6 K–10 K cycling levels, and 7 blocks were assigned to each of the 11 K–15 K cycling levels. This is summarized in Step 1 of Fig. 1.

For all codewords, three metrics were recorded after 10 cycles (`cycles_early`) and after 5,000 cycles (`cycles_late`). These metrics were codeword errors, page program time (for the page associated with that codeword), and block erase time (for the block associated with that codeword).

Finally, the number of errors per codeword was recorded when cycling completed.

4.2 Data Analysis

The probability density function (PDF) of errors per codeword across all cycling levels is shown in Fig. 2(a). Even though the endurance specification for this device is 5 K cycles at 100 bit ECC per codeword, Fig. 2(a) shows a mean of just 15 errors per codeword across all codewords tested from 6 K–15 K cycles. Just 7,857 codewords failed the 100 bit limit, out of 27,340,800 codewords tested in total. This gives a *codeword error rate* (CWER) of 0.028%.

Figure 2(b) shows the complementary cumulative distribution function (CCDF) for each cycling level, indicating the proportion of codewords from each cycling level that failed the 100 bit limit. The cycling level with the highest CWER was 12 K, with a CWER of just over 1E-3. No codewords failed after 6 K or 7 K cycles, highlighting the conservative nature of flash manufacturers' endurance specifications. While the errors per codeword generally increase with cycling level, the fact the cycling level with the highest CWER is 12 K, not 15 K, shows that tail codewords are not a function of cycling alone.

These results validate the hypothesis that the majority of sectors on a NAND device are capable of far exceeding the manufacturer's endurance rating. The next step in this research was to investigate if the measurements taken during the device's lifetime could predict the endurance capability of each sector, by generating a variety of alternative classification models.

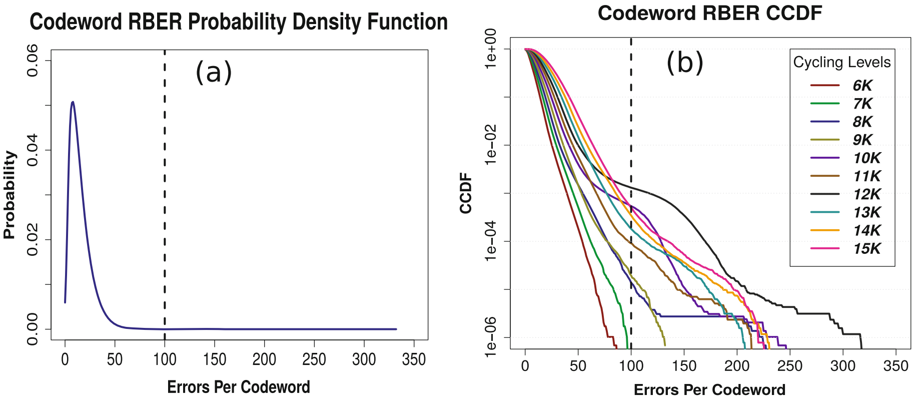


Fig. 2. (a) Probability Density Function of codeword errors for all codewords at end of cycling, and (b) Complementary Cumulative Distribution Function of codeword errors by cycling level at end of cycling.

5 Classification Models

5.1 Model Inputs and Outputs

As discussed in Sect. 2, a number of parameters were chosen as inputs to the model. The first of these is page number, because pages inherently have different error characteristics based on their position within a block. This means the page a codeword belongs to will provide important information about the likelihood of that codeword having higher or lower errors.

The next three input parameters are measurements taken after 10 cycles (the `cycles_early` cycling point, as discussed in Sect. 4.1). These are the program time of the page the codeword belongs to; the block erase time of the block the codeword belongs; and the number of errors in the codeword at this early point.

The next three input parameters are the same measurements taken after 5 K cycles (the `cycles_late` cycling point). Two points in life were chosen to allow the machine learning models to learn from the rate of change of these parameters.

The final input parameter is the cycling level the codeword was cycled to (6 K–15 K cycles).

The output of the classification model is whether the codeword passes or fails when cycling has completed. Less than or equal to 100 errors is deemed a pass; more than 100 errors is deemed a fail.

These inputs and outputs are summarized in Table 1.

In practice, inputs 1–7 would be measured for a codeword as the NAND device is being P-E cycled on an SSD. These would be supplied to the model, along with the cycling level of interest. The model would then predict if the codeword would pass or fail at that cycling level. For example, the inputs 1–7 would first be passed to the model together with a cycling level of 6 K. If the model predicted a pass, the same inputs would be passed with a cycling level of

Table 1. Data supplied to machine learning systems.

Parameter	Details
Input 1	Page number
Input 2	Program time at `cycles_early`
Input 3	Erase time at `cycles_early`
Input 4	CW errors at `cycles_early`
Input 5	Program time at `cycles_late`
Input 6	Erase time at `cycles_late`
Input 7	CW errors at `cycles_late`
Input 8	Cycling level
Output	0 if CW errors after cycling <= 100
	1 if CW errors after cycling >100

7 K, and so on until a cycling level is reached for which the model predicts a fail. The highest cycling level that gives a pass prediction is the predicted endurance level of the sector associated with that codeword.

5.2 Data Subsampling

As discussed in Sect. 4.2, the dataset of over 27 million codewords was heavily imbalanced, with just 0.03% of codewords failing at the decision boundary of 100 bits. Much work has been done on imbalanced datasets in recent years, with a popular approach being to subsample the majority class. This subsampling can be random [9] or informed [10].

For our initial runs, the majority class was subsampled randomly. However it was noticed that in some instances this resulted in the majority class having no samples close to the decision boundary of 100 bits (for example, 80 bits or lower). This meant the classifcation model would be suboptimal, as it could not learn about data points in this critical region.

We therefore propose a new type of informed subsampling called *PDF-based sampling*, in which the data is sampled according to the probability density function of the original data, as shown in Fig. 2(a). This technique ensures that the subsampled dataset has the same probability density function as the original dataset. This guarantees that the model will have full coverage of data points, and that each data point will be represented in proportion to its probability of occurence.

PDF-based sampling has similarities to a type of informed subsampling called stratified sampling [11]. Stratified sampling ensures that all areas of a population are represented in the subsampled data by dividing the population into groups called strata, and choosing samples from each stratum. PDF-based sampling could be considered a continuous case of this, such that every possible value within each stratum is represented.

For each of the machine learning methods employed in this study, models were generated and compared based on random subsampling and PDF-based subsampling.

5.3 Machine Learning Methods

The following eight machine learning methods were compared: Support Vector Machines (SVM), K-Nearest Neighbours, Decision Trees, Gradient Boosting, Random Forest, Neural Network, AdaBoost, and Naive Bayes.

The SVM runs were performed using the e1071 library. The SVM was first tuned on the data using the `R svm.tune` function to find the best values of the internal parameters `cost` and `gamma`.

All other machine learning runs were peformed using the `scikit-learn` machine learning library in `Python v2.7.`, with default internal parameters.

5.4 Classification Methodology

To compare the results of random versus PDF-based sampling, a hold-out test set was first removed from the data. This group contained 873 failing samples and 47,043 passing samples. Both passing and failing samples were selected using PDF-based sampling of their respective class. This ensured the hold-out test set was as representative of real-world data as possible.

Once the hold-out test set had been removed, 6,984 failing samples remained. A similar number of passing samples were chosen to make up the training set, and standard 8-fold cross validation was performed. Two training sets were built: one using random sampling of the passing samples and one using PDF-based sampling of the passing samples.

Models were evaluated on four classifier criteria: accuracy (Acc), sensitivity (Sn), specificity (Sp), and area under the ROC curve (AUC). Sensitivity is a measure of the true positive rate (TPR). For this research a positive was taken to be a passing codeword, so TPR is the proportion of passing codewords that are predicted to pass. Specificity is a measure of the true negative rate (TNR), or the proportion of failing codewords that are predicted to fail. Accuracy is the total proportion of correct predictions. The receiver operating characteristic (ROC) curve is obtained by plotting the TPR against the false positive rate (FPR) at various threshold settings. The area under the ROC curve is a summary measure of performance, that indicates the probability that a randomly chosen positive sample will be ranked higher than a randomly chosen negative sample.

In a strongly imbalanced dataset such as this, if the majority class was not subsampled it would be easy for the model to predict that every codeword would pass. In this case, the accuracy and the sensitivity would be excellent (99.97%, since 99.97% of all codewords pass), but the specificity would be poor. Therefore it is important the sensitivity and specificity are reasonably well balanced to ensure the model can predict both passing and failing codewords equally well.

The complete training datasets (from both random sampling and PDF-based sampling) were well balanced in terms of passing and failing codewords. However the data at each cycling level was not well balanced. For example, lower cycling levels contained more passing codewords and higher cycling levels contained more failing codewords. If the models were trained with cycling level as the only input, they would predict the majority class (pass or fail) for each cycling level.

The resultant accuracy, sensitivity and specificity figures were therefore used as the baseline for the model i.e. they represent how accurately the model would predict a passing or failing codeword across all cycling levels if the model was given no other input information.

6 Results

Table 2 shows the results for each of the eight machine learning methods employed. As mentioned in the previous section, results are presented for four criteria: accuracy, sensitivity, specificity and area under the ROC curve. For each machine learning method, average results from the 8-fold cross validation are provided (CV Average), plus the results from the hold-out test set trained on the entire training set (Hold-out Test). Two sets of results are provided: one for the randomly-sampled subset, and the other for the subset that used PDF-based sampling.

Table 2. Results table.

Method	Validation	Random Sampled				Proportional Sampled			
		Acc	Sn	Sp	AUC	Acc	Sn	Sp	AUC
Baseline	*CV Average*	*75.48*	*60.97*	*89.98*	*n/a*	*75.92*	*61.89*	*89.98*	*n/a*
	Hold-out Test	*60.85*	*60.32*	*89.46*	*n/a*	*60.85*	*60.32*	*89.46*	*n/a*
SVM	CV Average	99.35	99.20	99.50	**0.9977**	99.15	99.89	99.41	**0.9968**
	Hold-out Test	99.19	99.18	99.54	**0.9975**	99.24	99.23	99.54	**0.9980**
Nearest Neighbour	CV Average	98.80	98.62	98.80	0.9886	98.63	98.65	98.63	0.9856
	Hold-out Test	98.80	98.79	99.31	0.9905	98.98	98.98	98.85	0.9892
Decision Trees	CV Average	98.57	98.29	98.84	0.9855	98.38	98.44	98.33	0.9845
	Hold-out Test	97.49	97.45	99.43	0.9844	98.65	98.64	98.97	0.9881
Gradient Boosting	CV Average	**99.47**	**99.27**	**99.67**	0.9942	**99.27**	**99.13**	**99.41**	0.9926
	Hold-out Test	**99.25**	**99.24**	**99.89**	0.9956	**99.38**	**99.37**	**99.77**	0.9957
Random Forest	CV Average	99.07	99.00	99.15	0.9910	98.91	98.73	99.10	0.9898
	Hold-out Test	98.97	98.96	99.54	0.9925	99.14	99.14	99.43	0.9928
Neural Network	CV Average	93.83	90.50	97.14	0.9477	91.44	89.10	93.86	0.9324
	Hold-out Test	95.39	95.35	97.59	0.9647	97.99	98.02	96.33	0.9718
AdaBoost	CV Average	99.31	99.19	99.43	0.9932	99.11	99.13	99.10	0.9911
	Hold-out Test	99.20	99.19	99.43	0.9931	99.33	99.34	99.31	0.9932
Naive Bayes	CV Average	86.51	98.37	74.65	0.8650	85.17	97.99	72.31	0.8512
	Hold-out Test	97.94	98.35	76.17	0.8726	97.93	98.40	73.08	0.8574

The first row of the table shows the baseline results. As discussed in Sect. 5.4, these are the results that would be achieved if the classifier simply guessed the result for each cycling level, based on whether the training data contained more passing samples or failing samples for that cycling level.

It can be seen that SVMs, Gradient Boosting and AdaBoost all consistently achieve greater than 99% on all four criteria (Acc, Sn, Sp, AUC), across both methods of validation (CV Average and Hold-out Test) and both methods of sampling (random and PDF-based).

Nearest Neighbour, Decision Trees and Random Forest also perform well across all categories, with AUC values in excess of 0.98.

Neural networks perform less well, with an average AUC of 0.95 across both methods of validation and both methods of sampling.

Naive Bayes is the worst performing method, particularly in terms of specificity, achieving less than 80% across both methods of validation and both methods of sampling, making it unsuitable for this classification problem.

The best result in each category is highlighted in bold. It can be seen that Gradient Boosting achieves the highest accuracy, sensitivity and specificity across both methods of validation and sampling. However SVMs achieve the highest AUC (albeit slightly). Since the sensitivity/specificity values reported represent a single point on the ROC curve, this indicates that Gradient Boosting performs best at this point. However AUC is an average measure of performance over the whole TPR/FPR region, indicating that SVMs are a better general solution to this classification problem.

In terms of the two subsampling methods used (random and PDF-based), when tested on a common hold-out test set PDF-sampling results in marginally higher AUC. Closer analysis shows that sensitivity is better but specificity is worse, meaning that training on PDF-sampled data is better at classifying passing codewords but worse at classifying failing codewords. This is because PDF-based sampling ensures datapoints all the way up to the decision boundary, so that the dividing margin between both classes is too small, making it more difficult to identify failing codewords (negative class). This will be addressed in future research by placing more weight on the negative class.

7 Conclusions

This research gathered real large-volume data on the RBER of flash memory codewords across ten P-E cycling levels. More than 27 million codewords were tested in total. The aim was to build machine learning models that would predict if each sector of a NAND device would pass or fail at a given P-E cycling level, when supplied with performance data obtained as cycling progressed.

Eight different machine learning methods were evaluated and results were compared in terms of accuracy, sensitivity, specificity and area under the ROC curve. Three methods (SVMs, Gradient Boosting & AdaBoost) were found to give better than 99% performance across all evaluation critera, with SVMs being the best generalized solution.

Despite the extremely unbalanced nature of the dataset (only 0.03% of codewords failed across all cycling levels), the sensitivity versus specificity results were very well balanced. This means the models could correctly classify passing and failing codewords equally well.

The imbalanced data was managed by subsampling the majority class. As well as random sampling, a new sampling method called PDF-based sampling was proposed and investigated. PDF-based sampling was found to be an excellent technique for generating a hold-out test set as it guarantees the most realistic data possible for testing.

In addition, PDF-based sampling was shown to have potential for generating training data, giving higher accuracy, sensitivity and AUC than randomly sampled training data across all suitable machine learning methods identified, when tested on an independent test set. We believe this will be further improved in future research by focusing on difficult-to-classify negative samples using weighting.

In summary, results have shown that a number of machine learning methods may be used to predict the RBER of NAND flash codewords with exceptional accuracy. This finding has tremendous practical value to the flash and SSD industry, for whom the high RBER of modern flash devices is a key issue.

References

1. Bek, E.: Why your PC should have an SSD. In: 2015 Flash Memory Summit Conference, Santa Clara (2015). www.flashmemorysummit.com. Accessed 03 Aug 2017
2. Schroeder, B., Lagisetty, R., Merchant, A.: Flash reliability in production: the expected and the unexpected. In: 14th USENIX Conference on File and Storage Technologies (FAST 2016), pp. 67–80. USENIX Association, Santa Clara (2016)
3. Lee, J.D., Choi, J.H., Park, D., Kim, K.: Degradation of tunnel oxide by FN current stress and its effects on data retention characteristics of 90 nm NAND flash memory cells. In: Reliability Physics Symposium Proceedings, 2003. 41st Annual, 2003 IEEE International, pp. 497–501 (2003)
4. Mielke, N., Marquart, T., Wu, N., Kessenich, J., Belgal, H., Schares, E., Trivedi, F., Goodness, E., Nevill, L.R.: Bit error rate in NAND flash memories. In: 2008 IEEE International Reliability Physics Symposium, pp. 9–19 (2008)
5. Hogan, D., Arbuckle, T., Ryan, C.: Evolving a storage block endurance classifier for flash memory: a trial implementation. In: 2012 IEEE 11th International Conference on Cybernetic Intelligent Systems (CIS), pp. 12–17 (2012)
6. Arbuckle, T., Hogan, D., Ryan, C.: Learning predictors for flash memory endurance: a comparative study of alternative classification methods. Int. J. Comput. Intell. Stud. **3**(1), 18–39 (2014)
7. Jesd218a: Solid state drive (SSD) requirements and endurance test method. Standard, JEDEC (2011)
8. Mielke, N., Belgal, H., Fazio, A., Meng, Q., Righos, N.: Recovery effects in the distributed cycling of flash memories. In: Reliability Physics Symposium Proceedings, 2006. 44th Annual, IEEE International, pp. 29–35 (2006)
9. Batista, G.E.A.P.A., Prati, R.C., Monard, M.C.: A study of the behaviour of several methods for balancing machine learning training data. SIGKDD Explor. Newsl. **6**, 20–29 (2004)
10. Kubat, M., Matwin, S.: Addressing the curse of imbalanced training sets: one-sided selection. In: Proceedings of the Fourteenth International Conference on Machine Learning, pp. 179–186 (1997)
11. McKay, M.D., Beckman, R.J., Conover, W.J.: A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. In: Technometrics, American Society for Quality Control and American Statistical Association, pp. 55–61 (2000)

Opinion Extraction and Classification of Real-Time YouTube Cooking Recipes Comments

Randa Benkhelifa(✉) and Fatima Zohra Laallam

Department of Computer Science and Information Technologies,
Université Kasdi Merbah Ouargla, Route de Ghardaia, BP 511 30 000 Ouargla, Algeria
{randa.benkhelifa,laallam.fatima_zohra}@univ-ouargla.dz

Abstract. Applications based on Opinion Mining and Sentiment Analysis are critical tools for information-gathering to find out what people are thinking. It is one of the most active research areas in Natural Language Processing. In this paper, we develop a real-time system to extract and classify the YouTube cooking recipes reviews automatically. This system is based on Support vector machine approach and deals with the social media text characteristics. The proposed system collects data in real time from YouTube according to a user request. After it filters opinion texts from the other content, then it classifies it in (positive/negative) opinion. To improve the performance of our system we proposed some algorithms that constructed on sentiment bags, based on emoticons and injections.

Keywords: YouTube comments · Opinion mining · Support vector machine
Opinion extraction · Subjectivity · Emoticon · Injection

1 Introduction

Understanding emotions are one of the most important aspects of personal development and growth. Opinion Mining (OM) and Sentiment Analysis (SA) are the fields which study and analyze opinions, sentiments, and moods, based on textual data towards entities such as products, services, videos, etc. Exchanging recipes over YouTube has become popular over the last decade. It allows uploading recipes, searching for and downloading others, as well as to rate and review recipes. Sentiment analysis of food recipe comments is to identify what do people think about such cooking recipe video through users comments (positive or negative comments), where it is interesting to predict their ratings automatically.

Previous work [16] has shown that the reviews are the best rating predictors, in comparison to ingredients, preparation steps, and metadata.

This research proposes a new tool, which extracts YouTube cooking recipes comments automatically. Then classify them using a sentiment classification based on an approach that uses injections, emoticons and common words utilized in the cooking domain. This tool uses Natural Language Processing (NLP) to extract information from public reviews of YouTube videos, performing Sentiment Analysis to estimate the user preference associated with each recipe.

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 395–404, 2018.
https://doi.org/10.1007/978-3-319-74690-6_39

The details in this paper will be described in the following sections. The related works of sentiment analysis are explained in Sect. 2. The proposed method is outlined in Sect. 3. Next, the proposed system process, in Sect. 4, the results and discussion are demonstrated in Sect. 5. Finally, the conclusion of this research is summarized and presented in Sect. 6.

2 Related Works

Recently, several studies on opinion mining and sentiment analysis are crowned about comments in social media. Some research focuses on Subjectivity detection which can be defined as a process of selecting opinion containing sentences [13, 19]. The purpose of subjectivity and objectivity classification in opinion mining research is to distinguish between factual and subjective (expressing an opinion or emotion) remarks present in customer reviews [3, 6, 9, 20]. The authors in [1] aim to propose methods for identifying subjective sentences from customer reviews for mining product features and user opinions. Others research are interested in sentiment analysis which is a process of finding users opinion about the particular topic [22]. It performed on different domain data such as Movie [23], Books and Products [7, 10, 21], Restaurants [11], and cooking recipes [5], etc.

In [5] the authors have explored various strategies for predicting recipe ratings based on user reviews. Another related research about food recipe comments is the suggestion analysis for improving food recipes [15]. In [17] the authors present a sentiment based rating approach for food recipes which sorts food recipes present on various websites from sentiments of review writers. The work in [4] propose a menu generation system is described, that takes into account both user's preferences and healthy nutrition habits.

The emotions could be easily associated with an interesting application of human-computer interaction, where when a system identifies that the user is upset or annoyed, the system could change the user interface to a different mode of interaction as in [14]. In the works [15, 18] the authors have used a lexicon of the most used emoticons and injections.

In our case, we covered all that to obtain a complete system and used to make a decision.

3 Methodology

This section presents the methodology followed in this work. The objective of this work is to study the problem of short text special characteristics typically found on social media and to show how much it is important to consider these characteristics in the pre-processing phase.

YouTube comments are perfect for these due to their abundance and a short length. Moreover, YouTube is a popular video social network with a great diversity of users, which means that collecting a sufficiently large dataset with those characteristics on various topics is feasible. To ensure the consistency and the reliability of our proposed approaches, we tested our classifications and methods on a collection of 20000 recent texts of YouTube comments about videos of cooking recipes collected between (May and August 2016) from many YouTube videos. These texts were annotated manually by three human annotators as following:

- For creating the training model of the opinion filtering (5000 subjective and 5000 objectives)
- For creating the training model of the sentiment classification (5000 positive and 5000 negative).

All our proposed analysis for the both of classifications is shown in this section.

3.1 Bags Development

This step focuses on the informal language of the comments about cooking recipes video on YouTube. In this work, we have two types of bags were created: bag for emoticons and bag for interjections. After a deep analysis, we concluded the results, which are more than 60 emoticons and 130 injections. Some examples are shown in (Tables 1 and 2).

Table 1. Examples of the top used emoticons.

Positive emoticon	Negative emoticon
B-)	X-(
_	:-#
:*), :*	</3
:p, =p	O.o
8-)	:-(, :(
:), =), :-)	:_(
<3	:’(
XP, X-P	:/
XD, :D, =D, :-D	

Table 2. Examples of the top used injections

Positive injection	Negative injection
Wow, waw	Oh dear
Miam, miamy, miamiam	No way
Haha, hehe, hihi	Argh
Thank you	Boo, booh
Oy	Brr, brrr
Ahh, ahhh	Oops
Mmm, emm, hum, huumm	Bekhg, buk

Emoticons’ bag: We create a bag of the used emoticons in YouTube comments, whether they express a positive or a negative sentiment.

Injections’ bag: We create a bag of the injections in YouTube comments, whether they express a positive or a negative sentiment.

3.2 Algorithms Development

In this section, we show all algorithms developed and used to improve our classifications. The social media users utilize emoticons and injections in their social text. Knowing exactly the injection/emoticon used in the text is not important. The important is the sentiment reflected by those users. The classifier takes each emoticon or injection as a different word. But if we replace all (positive emoticon by PosEMO, positive injection by PosINJ, negative emoticon by NegEMO and negative injection by NegINJ) in sentiment classification, and all (emoticons by EMO and injections by INJ) in subjectivity classification. The classifier takes them as the same word (Tables 3 and 4).

Table 3. The developed algorithms for the subjectivity classification.

Algorithm1. Replace all injection with the same string "INJ"	Algorithm2. Replace all emoticon with the same string "IMO"
W ← Corpus I← set of Injections Foreach w∈W Foreach i ∈I If w= i w ←"INJ" EndIf EndForeach EndForeach	W← Corpus E ← set of Emoticons Foreach w ∈ W Foreach e ∈E If w = e w ← "EMO" EndIf EndForeach EndForeach

Table 4. The developed algorithms for the sentiment classification.

Algorithm3. Replace all positive injection by "PosINJ" and negative injection by "NegINJ"	Algorithm4. Replace all positive emoticon by "PosEMO" and negative emoticon by "NegEMO"
W ← Corpus PI ← set of positive Injections NI ← set of negative Injections Foreach w ∈ W Foreach pi ∈ PI If w = pi w ← "PosINJ" EndIf EndForeach Foreach ni ∈ NI If w =ni w ← "NegINJ" EndIf EndForeach EndForeach	W← Corpus PE← set of Positive Emoticons NE ← set of Negative Emoticons Foreach w ∈ W Foreach pe ∈PE If w = pe W ← "PosEMO" EndIf EndForeach Foreach ne ∈ NE If w = ne W ← "NegEMO" EndIf EndForeach EndForeach

3.3 Data Preprocessing and Feature Selection

The aim of this step is to clean the dataset by:

- Data pre-processing 1
  - Keeping stopwords;
  - Removing numbers and punctuations;
  - Removing all word appears less than 5 times;
  - Stemming: removing prefix and suffix finding the stem or the root of the word. The lovinsStemmer [12] is a well-known like stemming algorithm.
- Data pre-processing 2
  - Removing stopwords like ‘of’, ‘and’, ‘my’ that don’t have an influence on sentiment classification.
  - Removing numbers and punctuations.
  - Removing all word appears less than 3 times.
  - Stemming: removing prefix and suffix finding the stem or the root of the word. The lovinsStemmer [12] is a well-known like stemming algorithm.
- Feature selection
  - Using Term frequency inverse document frequency TF-IDF [2].
  - Part of Speech (POS) tagging: consists of tagging a word in a text to a particular part of speech based on its context and its definition. In English, it has nine parts of speech: noun, verb, article, adjective, preposition, pronoun, adverb, conjunction and interjection [15]. We use POS as features just for show which parts appear in which category more than the other.

3.4 The Training Model

The support vector machines (SVMs) have achieved great success in text classification, especially in the binary classification. In this paper, the both classifiers are building using Support Vector Machine (SVM) algorithm. SVMs are supervised learning models which analyze data and recognize patterns which are used for classification and regression analysis.

The proposed Approach

- For Opinion filtering classification, our approach is as follow:
  - V1: the original dataset Pre-Processed + TF-IDF feature selected.
  - V2: the original dataset pass firstly on Algorithm 1 then Algorithm 2, the result is Pre-Processed + TF-IDF feature selected.
- For the sentiment classification, our approach is as follow:
  - F1: the original dataset Pre-Processed + TF-IDF feature selected.
  - F2: the original dataset pass firstly on Algorithm 3 then Algorithm 4, the result is Pre-Processed + TF-IDF feature selected.

4 The Proposed System

This section describes the overall system (show Fig. 1).

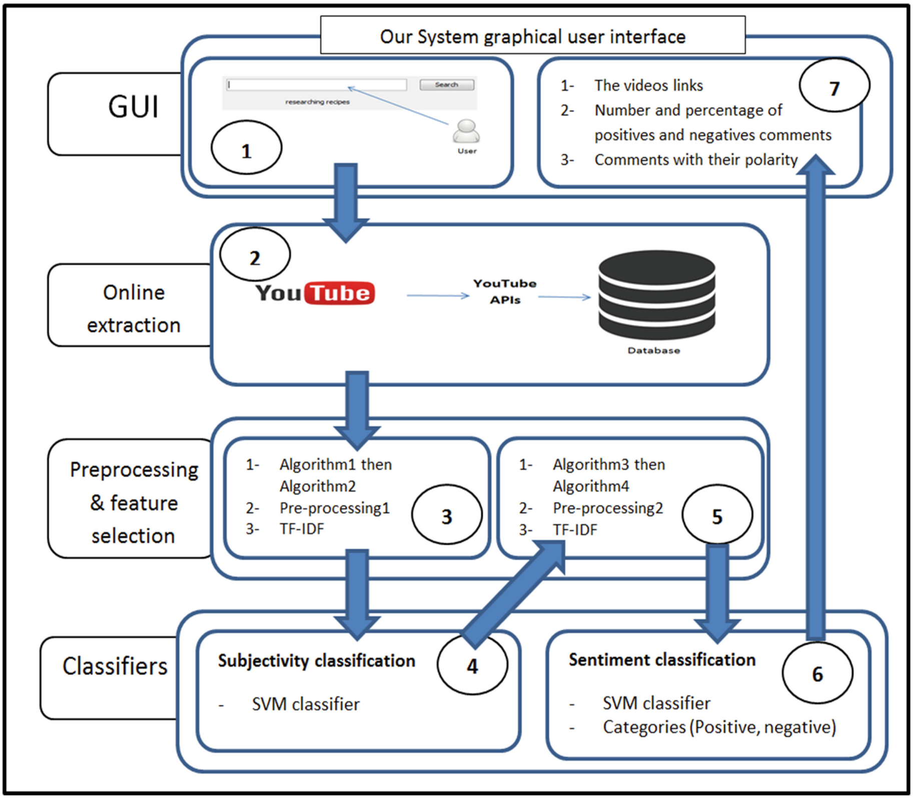

Our System graphical user interface
GUI
Search
researching recipes
User
1
1- The videos links
2- Number and percentage of positives and negatives comments
3- Comments with their polarity
7
Online extraction
2
YouTube
YouTube APIs
Database
Preprocessing & feature selection
1- Algorithm1 then Algorithm2
2- Pre-processing1
3- TF-IDF
3
1- Algorithm3 then Algorithm4
2- Pre-processing2
3- TF-IDF
5
Classifiers
Subjectivity classification
- SVM classifier
4
Sentiment classification
- SVM classifier
- Categories (Positive, negative)
6

Fig. 1. The proposed system

The proposed system works based on the following steps.

- Step 1: send the request: in our system users enter the cooking recipe name, and then click on the button search.
- Step 2: corpus collection: using the YouTube APIs "Google developers," our system will retrieve automatically the URLs of YouTube videos of the cooking recipe entered in the field. Then, our tool collects the generated comments on these recipes videos and stores them in a database.
- Step 3: pre-processing dataset and feature selection
  - Algorithm 1, then Algorithm 2,
  - Pre-processing 1,
  - Using TF-IDF Feature Selection.
- Step 4: Filtering opinions: in this step, the system filter automatically opinions "reviews" from the generated comments, and eliminating texts that bear no opinion by classifying the generated comments in the classes (opinion, other) using SVM classifier.
- Step 5: pre-processing dataset and feature selection
  - Algorithm 3 then Algorithm 4;

  - Pre-processing 2;
  - Using TF-IDF Feature Selection.
- Step 6: sentiment classification: classify the filtered opinions into pre-chosen classes as positive, or negative through the SVM classifier.
- Step 7: Getting the results: the system count and view the number of positive comments, and negative. It then displays the percentage and comments for each recipe video, and then it shows the overall percentage. And also our system users can see all the comments classified according to their polarity.

5 Results and Discussion

We have to experiment the two training models, using 10-fold cross-validation with WEKA [8] where SVM classifier is already implemented.

5.1 Experiment with the Opinion Filtering Training Model

We started our experiment with showing which parts of speech are the most appeared in each class (subjective and objective). The result is showing in Fig. 2.

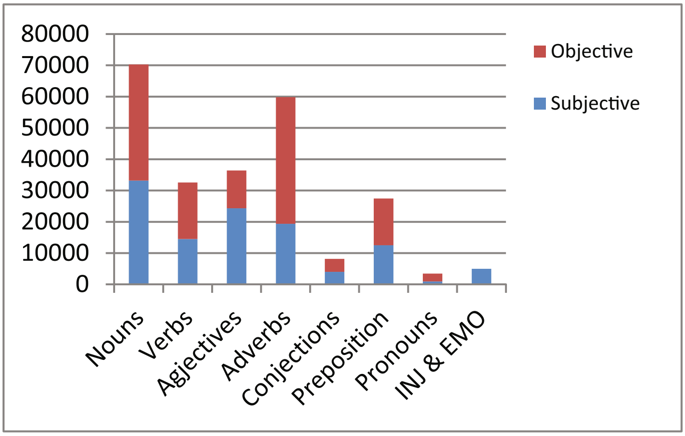


Fig. 2. The distribution of parts of speech in each category

According to Fig. 2, we note that injections and emoticons appear only in subjective text, which it contains more adjectives and adverbs. This due to its nature which refers to how someone's judgment is influenced by personal opinion and feeling, Contrasted with objective text which is related more to nouns, pronouns, verbs, and preposition to express existence and facts.

Using these characteristics of the subjectivity in the text (the appearance of injections and emoticons) in the subjectivity classification as Table 5 shows. The obtaining results with applying V1 data version then V2 (using Algorithm 1 then Algorithm 2) data version.

Table 5. Opinion filtering classification results

Version of data	SVM			
	Accuracy (%)	Recall	Precision	F-Measure
V1	78.7	0.787	0.787	0.787
V2	93.4	0.935	0.936	0.9355

Comparing the both results (see Table 5), the better results were got after applying the V2 data version. The improvement of the accuracy results using V1 and V2 is almost 14.7%. The same goes with the precision, recall and the F-measure. Table 6 shows the details of the results of each class (subjective and objective) using V2.

Table 6. Results based on V2 data version using Algorithm 1 and Algorithm 2

Version of data	SVM		
	Recall	Precision	F-Measure
Subjective	0.903	0.963	0.932
Objective	0.966	0.909	0.937

Table 6 calculates the performance results for the classification of the binary classifier at the stage of using V2.

5.2 Experiment with the Sentiment Training Model

Table 7 shows the obtaining results with applying F1 data version then F2 (using Algorithm 3 and Algorithm 4) data version.

Table 7. Sentiment classification results

Version of data	SVM			
	Accuracy (%)	Recall	Precision	F-Measure
F1	83.5	0.835	0.835	0.835
F2	95.3	0.953	0.953	0.9535

Regarding the effect of using Algorithm 3 then Algorithm 4 on the sentiment classification performance, we can note that there was an improvement of 11.8% in the accuracy, and also in the recall, precision and the F-Measure (Table 7).

Table 8 shows the performance results for the classification of the binary classifiers at the stage of using F2.

Table 8. Results based on F2 data version using Algorithm 3 and Algorithm 4

Version of data	SVM		
	Recall	Precision	F-Measure
Positive	0.964	0.944	0.954
Negative	0.943	0.963	0.953

6 Conclusion

This paper proposes a system for opinion extraction and classification automatically in real time from YouTube on cooking recipes comments. Firstly our system collects comments about cooking recipes videos. Next step is to filter undesired texts (objective texts) using SVM classifier which has 93.4% of accuracy. Then to classify this subjective texts (opinions) into a positive or negative class using the model built by SVM classifier which has 95.3% of accuracy. Our system can classify cooking recipes into ‘‘recommended” or ‘‘not recommended” and also compare between recipes.

References

1. Kamal, A.: Subjectivity classification using machine learning techniques for mining feature opinion pairs from web opinion sources, New Delhi, India (2013)
2. Sebastiani, A.: Machine learning in automated text categorization. ACM Comput. Surv. **34**, 1–47 (2002)
3. Pang, A., Lee, L.: A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In: Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL 2004), Barcelona, ES, pp. 271–278 (2004)
4. Bianchini, D., De Antonellis, V., De Franceschi, N., Melchiori, M.: PREFer: a prescription-based food recommender system. Comput. Stand. Interfaces **54**, 64–75 (2017)
5. Liu, C., Guo, C., Dakota, D., Rajagopalan, S., Li, W., Kübler, S.: My curiosity was satisfied, but not in a GoodWay: predicting user ratings for online recipes. In: Proceedings of the Second Workshop on Natural Language Processing for Social Media (SocialNLP), Dublin, Ireland, 24 August 2014, pp. 12–21 (2014)
6. Chaturvedi et al.: Bayesian network based extreme learning machine for subjectivity detection. J. Frankl. Inst. (2017). https://doi.org/10.1016/j.jfranklin.2017.06.007
7. Dave, K., Lawrence, S., Pennock, D.: Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In: Proceedings of the 12th International Conference on World Wide Web, WWW 2003. ACM, New York (2003)
8. Witten, H.A., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. Morgan Kaufmann, San Francisco (2000)
9. Höpken, W., Fuchs, M., Menner, T., Lexhagen, M.: Sensing the online social sphere using a sentiment analytical approach. In: Xiang, Z., Fesenmaier, D. (eds.) Analytics in Smart Tourism Design, pp. 129–146. Springer International Publishing, Cham (2017)
10. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2004. ACM, New York (2004)
11. Liu, J., Seneff, S., Zue, V.: Harvesting and summarizing user-generated content for advanced speech-based HCI. IEEE J. Sel. Topics Sig. Process. **6**(8), 982–992 (2012)
12. Lovins, J.B.: Development of a stemming algorithm. Mechanical Translation and Computational Linguistics (1968)
13. Durant, K.T., Smith, M.D.: Mining sentiment classification from political web logs. In: WEBKDD 2006, Philadelphia, Pennysylvania, USA. ACM (2006). 1-59593-4448
14. Hankin, L.: The effects of user reviews on online purchasing behavior across multiple product categories. Master’s final project report, UC Berkeley School of Information (2007)

15. Pugsee, P., Niyomvanich, M.: Suggestion analysis for food recipe improvement. In: Proceeding of the 2015 International Conference on Advanced Informatics: Concepts, Theory and Application (ICAICTA) (2015)
16. Yu, N., Zhekova, D., Liu, C., Kübler, S.: Do good recipes need butter? Predicting user ratings of online recipes. In: Proceedings of the IJCAI Workshop on Cooking with Computers, Beijing, China (2013)
17. Rao, S., Kakkar, M.: A rating approach based on sentiment analysis. In: 2017 7th International Conference on Cloud Computing, Data Science & Engineering-Confluence, pp. 557–562. IEEE, January 2017
18. Hamouda, S.B., Akaichi, J.: Social networks' text mining for sentiment classification: the case of Facebook' statuses updates in the 'Arabic Spring' Era. Int. J. Appl. Innov. Eng. Manag. (IJAIEM), **2**(5), 470–478 (2013)
19. Verma, S., Bhattacharyya, P.: Incorporating semantic knowledge for sentiment analysis. In: Proceedings of International Conference on Natural Language Processing (2009)
20. Wilson, T.: Fine-grained subjectivity and sentiment analysis: recognizing the intensity, polarity, and attitudes of private states. University of Pittsburgh (2008)
21. Tan, S.S., Na, J.C.: Mining semantic patterns for sentiment analysis of product reviews. In: International Conference on Theory and Practice of Digital Libraries, pp. 382–393. Springer, Cham, September 2017
22. Raut, V.B., et al.: Survey on opinion mining and summarization of user reviews on web. Int. J. Comput. Sci. Inf. Technol. (IJCSIT) **5**(2), 1026–1030 (2014)
23. Zhang, X., Zhu, F.: The influence of online consumer reviews on the demand for experience goods: the case of video games. In: 27th International Conference on Information Systems (ICIS), Milwaukee. AISPress (2006)

A Framework for Big Data Analysis in Smart Cities

Hisham Elhoseny[1(✉)], Mohamed Elhoseny[1,3], A. M. Riad[1], and Aboul Ella Hassanien[2,3]

[1] Faculty of Computers and Information, Mansoura University, Mansoura, Egypt
h.elhoseny88@yahoo.com, mohamed_elhoseny@mans.edu.eg
[2] Faculty of Computers and Information, Cairo University, Giza, Egypt
[3] Scientific Research Group in Egypt (SRGE), Cairo, Egypt

Abstract. Due to the rapid change in technologies, new data forms exist which lead to a huge data size on the internet. As a result, some learning platforms such as e-learning systems must change their methodologies for data processing to be smarter. This paper proposes a framework for smoothly adapt the traditional e-learning systems to be suitable for smart cities applications. Learning Analytics (LA) has turned into a noticeable worldview with regards to instruction of late which embraces the current progressions of innovation, for example, cloud computing, big data processing, and Internet of Things. LA additionally requires a concentrated measure of preparing assets to create applicable investigative outcomes. Be that as it may, the customary methodologies have been wasteful at handling LA difficulties.

Keywords: Big data · E-learning · Smart learning · Smart systems · Smart cities Learning Analytics · Internet of Things

1 Introduction

Smart Cities provide the opportunity to integrate the physical infrastructures of the city such as the transportation sector, utilities, land, and city services. The smart city model typically integrates the economic, social and environmental components of the city in a way that sustainably maximizes the efficiency of the city's primary systems. Rolimetal (2015) defined the Smart City as an urban system that utilizes ICTs to develop infrastructures and public services that are in a more accessible, interactive and efficient context. Fundamentally, Information and Communication Technologies (ICTs) and geographic information technologies (GIS) have revolutionized the communications and interaction capabilities across urban settings in the form of high level decision making tools [1].

The continuous technology development empowers learners to take in more viable, proficiently, adaptable and serenely. Smart learning, an idea that portrays learning in advanced age, has increased and expanded in different applications. This paper discusses and clarifies the technical meaning of Smart learning and proposes an applied structure towards building such learning environment. The smart teaching method structure

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 405–414, 2018.
https://doi.org/10.1007/978-3-319-74690-6_40

incorporates class-based separated direction [2], gather based communitarian learning, individual-based customized learning and mass-based generative learning.

The Smart learning system utilizes IoT as a component of the advanced development expected to associate with the client for personalization and provides a new customized learning tools. Versatile processing advancements are utilized to associate with the users, sense their constant exercises, and give them whenever anyplace open door for smart learning system. In addition, IoT tools are recently used with big data based application to provide to manage the difficulties related to the data size, the processing speed, etc. [3].

Accordingly, we propose a Smart Learning Framework that provides a conceptual model for Big Data learning Analytics platform. The reset of the paper is organized as follows. In Sect. 2, we discuss the Related Work of applying the smart learning in Big Data and Internet of Things applications. In Sect. 3, a comparative study of the existing smart learning systems is discussed. In Sect. 4, the proposed model of Smart Learning Systems is explained. In Sect. 5 Implementation of Analytics and the results of the proposed model are discussed. Finally, Sect. 6 concludes the paper.

2 Related Work

The idea of a smart city itself is as yet rising, and crafted by characterizing and conceptualizing it is in advance [6, 11]. The idea is utilized everywhere throughout the world with various terminologies, setting and implications. A scope of calculated variations created by supplanting the word shrewd with descriptive words for example, advanced or smart are promptly utilized and reused. Some are perceiving the utilization of smart city as a urban naming marvel [11], noticing that the name smart city is an idea and is utilized as a part of ways that are not generally reliable. A few working definitions have been advanced and received in both useful and scholarly utilize. This dissonance of definitions is bringing about calls for theoretical research in such manner [6], a smart city signifies an instrumented, interconnected, and clever city. Instrumentation empowers the catch and reconciliation of live genuine information using sensors, booths, meters, individual gadgets, apparatuses, cameras, advanced mobile phones, embedded medicinal gadgets, the web, and other comparative information securing frameworks, including interpersonal organizations as systems of human sensors. Interconnection implies the reconciliation of those information into a venture figuring stage and the correspondence of such data among the different city administrations. Knowledge alludes to the consideration of complex examination, demonstrating, improvement, and representation in the operational business procedures to improve operational choices. Conversely, the Natural Resources Defense Chamber [9] characterizes more astute in the urban setting as more effective, manageable, fair, and reasonable. Toppeta [9] stresses the change in maintainability and bearableness. Washburn et al. [9] see a smart city as a gathering of smart registering innovations connected to basic framework parts and administrations. Smart figuring alludes to a new era of incorporated equipment, programming, and organize advances that give IT frameworks and real time consciousness of this present reality and progressed examination and activities that enhance business forms [9].

Learning Analytics is for the most part concerned with the instructive issues and student achievement. LA uses strategies in gathering information from students, investigating information and extricating profitable data from them, and revealing the results to the learner, teacher and the establishment. A definitive objective of LA is to grow better approaches to break down instructive information what's more, continually enhance the learning and educating forms [8]. It goes for changing the instructive information into valuable activities to upgrade the nature of learning [12].

3 Analysis of Existing Smart Learning Paradigms

Cutting edge smart learning structure is typically incorporated with different parts of new smart learning frameworks with the essential smart learning substance. Instruction innovation like learning administration Systems (Moodle, Blackboard and so on.), different equipment and programming apparatuses give the stretched out office to data transmissions and collaborations [4]. Combination of different examination and assessment apparatuses with in the learning administration, with the assistance of effective interfaces empowers execution assessment of the students in this manner serves to examinations, how much learning has happened? [5]. Innovation progression away methods, accessibility of new and minimal capacity devices guaranteed capacity of complex information and data regardless of size, volume and qualities. Complex smart learning information additionally comprises of mixed media data like sound, video, schematics, content, activities, 3d models, which typically possesses gigantic spaces. Getting to and verification technique fuse in smart learning structure encouraged specific and wanted data trade between students. Combination of information administration and database devices joins and deal with the e-learning data utilizing content administration systems [6].

Improvement of smart learning paradigm with authoritative also, administration abilities, following, execution assessment methods and other new highlights inferred new learning frameworks like Learning administration System. Likewise, sharable content object reference model (SCROM), installing client creating capacities, consolidation of customization, content composing, UI writing, cooperative learning, customizer UI highlights, additionally reinforce the e-learning system [8] as in Table 1.

4 The Proposed Smart Learning Framework

As shown at Fig. 1, smart learning is an important component towards building a smart city which requires a smart big data processing and transmission [1, 7].

However, big data framework for smart learning starts with recognizable proof of information source and can be characterized as an information system. This information must accessible from all smart learning sources including LMS, CMS, File frameworks, Web, Social Media. Extraction structure removes the information from information

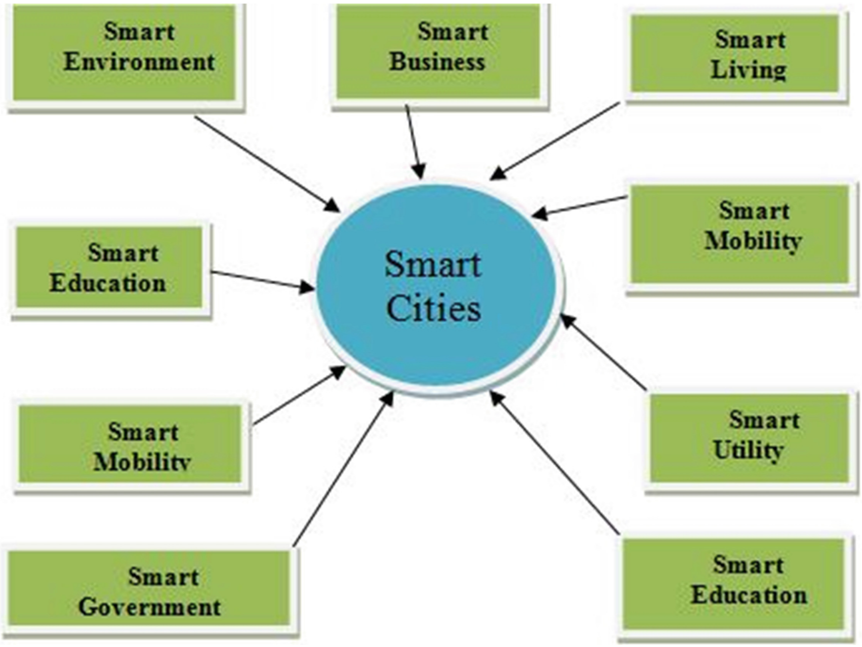


Fig. 1. Smart system framework components.

Table 1. Evaluation of features in smart learning system

Smart learning components	SLMS	Smart Virtual classroom	SCMS	SILMS
Performance evaluation	✓	✗	✗	✓
Multimedia	✓	✓	✓	✓
Interface authoring	✗	✗	✗	✓
Tracking feature	✓	✗	✓	✓

structure utilizing scientific classification, joint effort, faceting, labeling insight extraction. When extraction is finished, expository system starts for semantic handling, cosmology contemplate, grouping, pertinence think about, thesauri, system object development etc. [2]. Processing the appropriate data structure starts after searching, ordering, creeping, changing over of information after the initial data collection. When information handling is done, big data application will be ready for use in a smart environment. Accordingly, inquiry, setting creation, question directing will be available in the application.

The proposed smart learning framework combines both of e-learning worldview with the advantage of big data analysis. The proposed system coordinates three layers of various innovation structures.

Smart learning framework consists of smart information transmission layer, smart devices layer and smart application layer as shown in Fig. 2. Together, these layers provide data transmission amongst a learner and the framework [12].

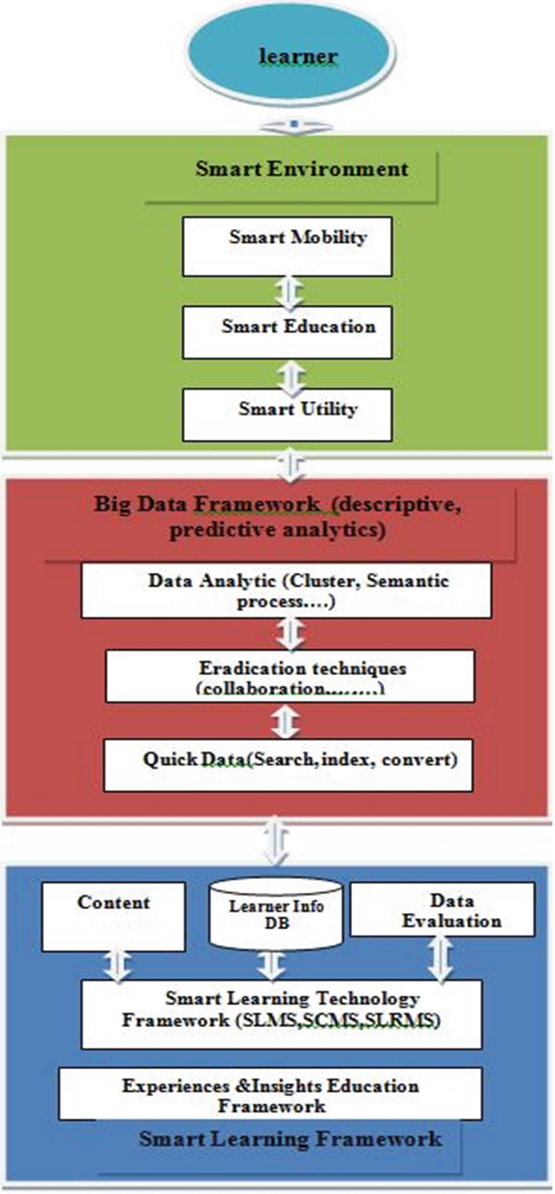

learner
Smart Environment
Smart Mobility
Smart Education
Smart Utility
Big Data Framework (descriptive, predictive analytics)
Data Analytic (Cluster, Semantic process....)
Eradication techniques (collaboration........)
Quick Data(Search,index, convert)
Content
Learner Info DB
Data Evaluation
Smart Learning Technology Framework (SLMS,SCMS,SLRMS)
Experiences &Insights Education Framework
Smart Learning Framework

Fig. 2. Proposed smart learning framework

Learning Analytics for Big Data contains three diagnostic standards: descriptive, predictive and prescriptive analytics. These three standards are interconnected with each other and trade distinctive information things. We continue with one specific case to expand their capacities. Consider questioning all Learners who have a specific level of C# programming dialect expertise. Given understudies' imprints in the range from 0 to 100, they ought to be arranged in one of the four score scales to fulfill our target. The four score scales can be characterized as HU for high uniqueness in the range from 75 to 100, VG for qualification from 65 to 74, G for credit from 50 to 64 and W from 0 to 49 for come up short. We will likely recognize understudies at the danger of being flopped in the C# programming course and give them smart criticism to bring them back on track and lead them to pass the course [8, 10].

5 The Proposed Model Implementation

Like any information systems project, implementing Big Data Analytics project within an organization has its special challenges. Putting the required infrastructure in place, higher initial cost, changes to business processes [13] and availability of experienced data scientists – these are some of the challenges in implementation of Analytics within an organization.

Penetration of internet of things, smart phones and cloud computing technologies in the industry and society will continue to spawn even higher levels of Big Data. To leverage insights from Big Data, industry and academia will be in need of experienced professionals in Data Science and Predictive Analytics. This field requires both domain knowledge and broad set of quantitative skills such as Statistics, Forecasting, Optimization, simulation, probability etc. [14]. Training thousands of data-scientists and then translating those into measurable business outcomes will remain a challenge for academia and industry [15].

Dealing with Big Data from variety of sources (organizational legacy system, ERP, social media) and deriving value from it, requires a clear strategy and implementation. Data Scientists with knowledge, experience and track record need to be on board. Initial projects must be monitored from top management. After successful implementation of Big Data strategy for a pilot site, opportunities for the business context can be identified for rest of the organization [7].

The descriptive analytics (DA) will compress the instructive data of all learners enlisted in the software engineering courses and channels the individuals who have passed the C# programming course or have presently taken the course. As DA is focused on the historical data in the past, it produces investigates the quantity of Learners who have passed the course with HD mark. This data will be given to the DA. This information will be given to the descriptive analytics by the Contextualization Server.

The predictive analytics (PA) take into consideration learners who have officially taken the C# programming course. It uses exact machine learning procedures to extrapolate the probability of those learners being classified in one of the four review scales toward the finish of the semester in view of their past execution in tests/assignments. The consequence of the prescient examination stage is extrapolation drifts through which learner's execution is delineated trailed by their likelihood scores. In particular, the patterns will create every learners imaginable future results regarding their last checks with the probability of each mark. For instance, the prescient stage figures four distinctive review scales likely situations for learner 1 alongside their likelihood scores. This data can be used to recognize in danger of being fizzled Learners. Figure 3 illustrates one sample result of the PA phase in projecting Learner 1 and Learner 2 final marks classified in four review scales with their comparing likelihood scores. As p Fig. 3, learner 1 is probably not going to fall flat the course given that with 75% likelihood they will pass. Notwithstanding, learner 2 falls in the classification of in danger of coming up short the course with the anticipated likelihood of projected probability of falling at 64%. Figure 4 delineates the root mean square error (RMSE) between the anticipated and real esteems Learners.

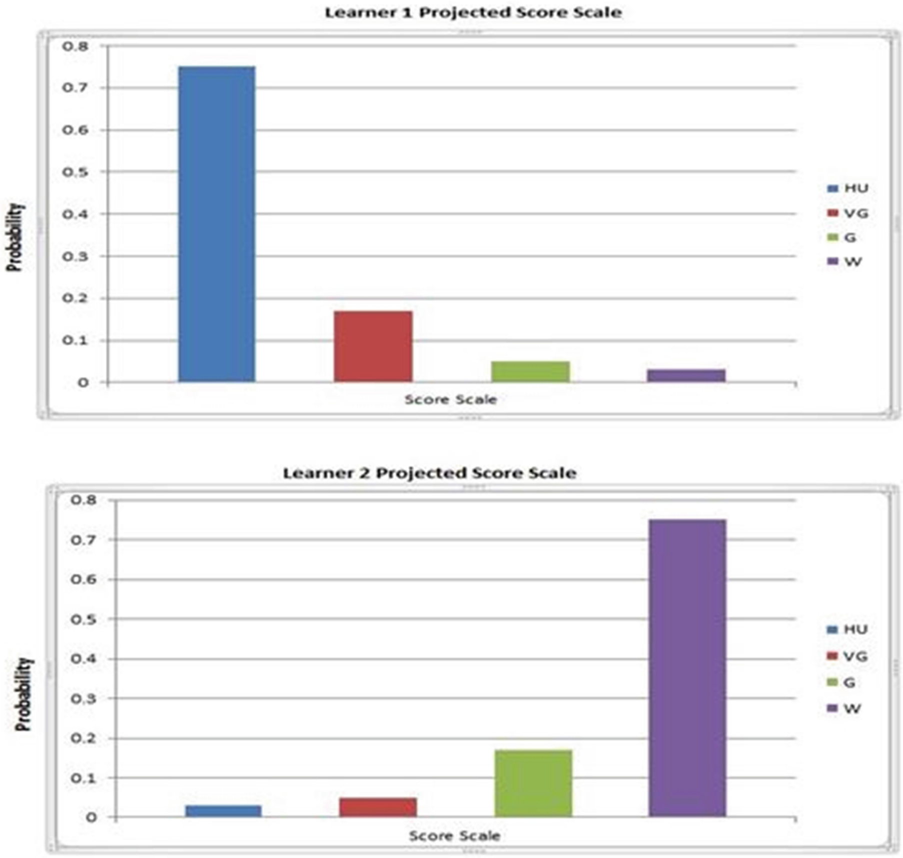


Fig. 3. Predictive analytics projected score scales for learner 1, 2

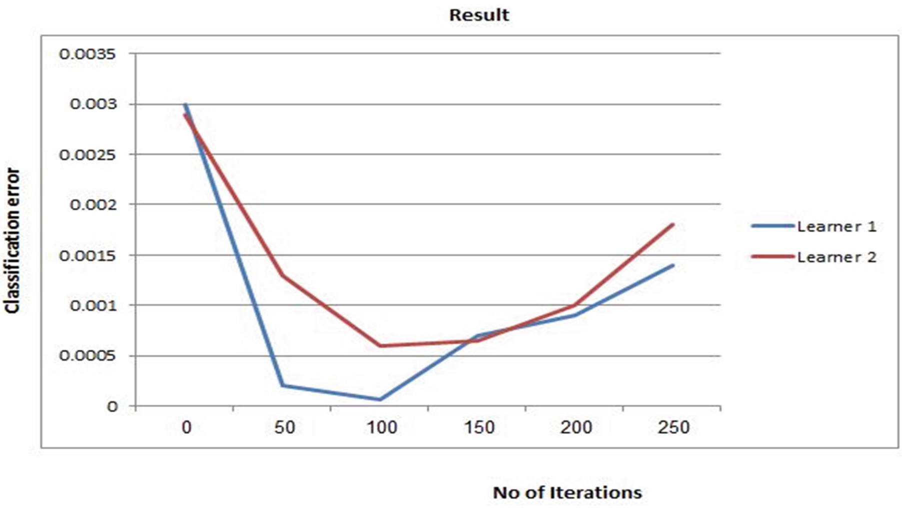


Fig. 4. RMSE shows the differences between predictive analytics projected score scales for learner 1, 2

The PA mulls over the predefined objective as the rundown of in danger of falling flat learners in the C# programming course. At that point it applies specific suggestion and reenactment systems to produce certain courses of activities for the objective learners to enable them to pass the course. These activities can be of various types, for example, proposing further learning materials/assets to be contemplated, prescribing specific tests covering misjudged ideas to be taken, and endorsing unique mentoring labs, then interview sessions to go to. As shown in Table 2 Predictive Analytics Projected Score Scales for Learner 1, 2.

Table 2. Predictive score analytics projected scales for Learner 1, 2

Score scale	Learner	
	Learner 1	Learner 2
HU	0.75	0.05
VG	0.15	0.07
G	0.07	0.15
W	0.05	0.75

6 Conclusion

This paper proposes a framework for smoothly adapt the traditional e-learning systems to be suitable for smart cities applications. Smart learning structure, when created has the potential to provide the needs of learning condition not just with the learning asset, yet in addition with the learner's information as well. Smart learning systems like versatile learning, aptitude based learning, venture based learning, task based learning, flipped learning are investigated till date, to give better learning strategies. However, none of these are having the in-depth design acknowledgments of Learners data or grouping of learning designs, which is a standout amongst the most required variables for cutting edge increased learning in a big data and smart applications environments. Dependently, this paper proposed a smart learning framework which is appropriate for working in smart environments with big size of data. The proposed framework is analyzed and extensively discussed. A sample of the implementation results are also discussed.

References

1. Elhoseny, H., Elhoseny, M., Abdelrazek, S., Bakry, H., Riad, A.: Utilizing Service Oriented Architecture (SOA) in smart cities. Int. J. Adv. Comput. Technol. (IJACT) **8**(3), 77–84 (2016)
2. Shehab, A., Elhoseny, M., Hassanien, A.: A hybrid scheme for automated essay grading based on LVQ and NLP techniques. In: Proceedings of 12th International Computer Engineering Conference (ICENCO), pp. 65–70. IEEE (2016). https://doi.org/10.1109/ICENCO.2016.7856447
3. Digolo, B.A., Andang'o, E.A., Katuli, J.: E-learning as a strategy for enhancing access to music education. Int. J. Bus. Soc. Sci. **2**(11), 135–139 (2011)
4. Shehab, A., Ismail, A., Osman, L., Elhoseny, M., El-Henawy, I.M.: Quantified self using IoT wearable devices. In: The 3rd International Conference on Advanced Intelligent Systems and Informatics (AISI2017), 9–11 September 2017, Cairo-Egypt. Springer (2017)
5. Kumar, P.: Big data integration for transition from e-learning to smart learning framework. In: 2016 3rd MEC International Conference on Big Data and Smart City (2016)
6. Boulton, A., Brunn, S.D., Devriendt, L.: Cyberinfrastructures and "smart" world cities: physical, human, and soft infrastructures. In: Taylor, P., Derudder, B., Hoyler, M., Witlox, F. (eds.) International Handbook of Globalization and World Cities. Edward Elgar, Cheltenham (2011). http://www.neogeographies.com/documents/cyberinfrastructure_smart_world_cities.pdf
7. Elhoseny, H., Elhoseny, M., Abdelrazek, S., Riad, A.M.: Evaluating learners' progress in smart learning environment. In: The 3rd International Conference on Advanced Intelligent Systems and Informatics (AISI2017), 9–11 September 2017, Cairo-Egypt. Springer International Publishing AG (2018). https://doi.org/10.1007/978-3-319-64861-3_69
8. Metawa, N., Elhoseny, M., Hassanien, A.: An automated information system to ensure quality in higher education institutions. In: Proceedings of 12th International Computer Engineering Conference (ICENCO), pp. 196–201. IEEE (2016). https://doi.org/10.1109/ICENCO.2016.7856468
9. Toppeta, D.: The smart city vision: how innovation and ICT can build smart, "Livable", sustainable cities. The Innovation Knowledge Foundation (2010). http://www.thinkinnovation.org/file/research/23/en/Toppeta_Report_005_2010.pdf
10. Siemens, G., Long, P.: Penetrating the fog: analytics in learning and education. EDUCAUSE Rev. **46**(5), 30 (2011)
11. Harshawardhan, S., Devendra, P.: A review paper on big data and hadoop. Int. J. Sci. Res. Pub. **4**(10), 1 (2014). ISSN 2250-3153
12. Wilairat, Y., Thara, A., Jitimon, A.: SQL learning object ontology for an intelligent tutoring system. Int. J. e-Education e-Business e-Management e-Learning **3**(2) (2013)
13. Bose, R.: Advanced analytics: opportunities and challenges. Ind. Manag. Data Syst. **109**(2), 155–172 (2014)

14. Waller, M.A., Fawcett, S.E.: Click here for a data scientist: big data, predictive analytics, and theory development in the era of a maker movement supply chain. J. Bus. Logist. **34**(4), 249–252 (2013)

15. Dyche, J.: Big data and discovery (2012). http://jilldyche.com/2012/12/04/big-data-and-discovery/. Accessed 12 Dec 2012

Performance Enhancement of Distributed Clustering for Big Data Analytics

Omar Hesham Mohamed[1(✉)], Mohamed Elemam Shehab[2], and Essam El Fakharany[3]

[1] Information System Department, Arab Academy for Science,
Technology & Maritime Transport, Cairo, Egypt
Omar.hesham.shehab@gmail.com

[2] Arab Academy for Science, Technology & Maritime Transport, Cairo, Egypt
melemam9@gmail.com

[3] College of Computing and Information Technology, Arab Academy for Science,
Technology & Maritime Transport, Cairo, Egypt
essam.elfakharany@aast.edu

Abstract. Big Data analytics are recently coming up as prominent research area in the field of data science. Apache Spark is an open source distributed data processing platform that uses distributed memory abstraction to process large volume of streaming data efficiently. Performance improvement of analytic computational model of streaming big data is important to meet the requirements of many real-time data analysis. Researchers focus on Analytic algorithm improvement to reduce analysis time. This paper presents performance enhancement of in-memory computational model by selecting the most important attributes after caching data to Apache spark. Performance analysis of distributed K-Means clustering algorithm based on in-memory computational model has been conducted. The results show improvement in the performance of the model.

Keywords: Big Data · Apache Spark · Machine learning algorithms
K-Means algorithm · In-memory computation · Big data analytic

1 Introduction

The world today is built on the foundations of data. Lives today are impacted by the ability of the companies to dispose, interrogate and manage data. The development of technology infrastructure is adapted to help generate data, so that all the offered services can be improved as they are used. As an example, internet today became a huge information-gathering platform due to social media and online services. At any second, there are new added data [1]. The explosion of data cannot be any more measured in gigabytes; since data is bigger there are used exabytes, zettabytes and yottabytes [2].

In order to manage this huge amount of data stored, we have started to hear the expression of "Big Data" in order to understand this expression. We have first to know that a study on the Evolution of Big Data as a Research and Scientific Topic shows that the term "Big Data" was present in research starting with 1970s but has been comprised in publications in 2008 [3]. It stands to reason that in the commercial sector Big-Data

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 415–425, 2018.
https://doi.org/10.1007/978-3-319-74690-6_41

has been adopted more rapidly in data driven industries, such as financial services and telecommunications, which it can be argued, have been experiencing a more rapid growth in data volumes compared to other market sectors, in addition to tighter regulatory requirements and falling profitability. At first, Big Data was seen as a mean to manage to reduce the costs of data management. Now, the companies focus on the value creation potential. In order to benefit from additional insight gained there is the need to assess the analytical and execution capabilities of "Big Data".

Big Data is the collection of digital records collected by various business and government organizations using not only simple traditional information exchange software and hardware such as computer, laptops, mobile phone etc. but also from technically complex sources such as embedded systems which are collection of various sensors used for various purposes like for city streets (cameras, microphones) or jet engines (temperature sensors) etc. as well as from IOTs (Internet of Things) which is used for data collection of electrical device data over internet [4–6].

In order to obtain a precise definition of what Big Data was, it was useful to define Big Data by using an equation and then classifying the various types of Big Data according to the equation. Big Data (BD) consisted of structured Big Data (SBD) which existed typically in relational database rows and columns and unstructured Big Data (UBD) which did not exist usually in relational database rows and columns. Big Data could then be defined by the following equation:

$$\mathrm{BD} = \mathrm{SBD} + \mathrm{UBD}.$$

This massive growth generates the need of exploring appropriate tools, techniques and frameworks related to big data storage, computation, security, privacy, and analytics under the consideration of velocity, volume, and variety involved in it. These issues are hard to handle by traditional approaches because of the following six-dimensional taxonomies which are Data, Analytics, Security and privacy, Visualization, Compute infrastructure and Storage infrastructure [7, 8].

Big Data could be classified as structured and unstructured data which organizations of any type handled on a daily basis and which could not be governed and mined for useful information by current information technology tools. Unstructured data was characterized as data that did not appear in relational database rows and columns [9].

The scalability of big data solutions within data centers is an essential consideration. Data is vast today, and it is only going to get bigger. If a data center can only cope with the levels of data expected in the short to medium term, businesses will quickly spend on system refreshes and upgrades. Forward planning and scalability are therefore important [2].

Apache spark uses machine learning library (MLLib) which has been used in this paper so first we will go in brief to the techniques of machine learning. There are three common categories of machine learning techniques are Classification, Clustering and Collaborative Filtering [10].

Classification: Gmail uses a machine learning technique called classification to designate if an email is spam or not, based on the data of an email: the sender, recipients, subject, and message body. Classification takes a set of data with known labels and learns how to label new records based on that information.

Clustering: Google News uses a technique called clustering to group news articles into different categories, based on title and content. Clustering algorithms discover groupings that occur in collections of data.

Collaborative filtering: Amazon uses a machine learning technique called collaborative filtering (commonly referred to as recommendation), to determine which products users will like based on their history and similarity to other users.

2 Big Data Computational Model

Big Data computational models need appropriate algorithms and frameworks for managing and processing huge structured/unstructured data generated from various sources. Traditional models are having limited capabilities therefore a completely new infrastructure is needed for management and processing of big data [3, 11]. In couple of years, there is tremendous growth seen in development of big data computation models, Big data computing infrastructure is divided mainly into two branches which are stream data processing which we are going to discuss in this research about a way of enhancing one of its tools and batch data processing. For stream data processing Infosphere (IBM tool for data analysis), Storm and Spark (both are two tools for Apache used for data analysis) are available [12]. Apache Spark is taken in this research work for analysis as we are going to focus on in-memory computation and we propose a way to enhance the output by decreasing the time without changing the efficiency of the algorithm used [13].

For In-memory based computational model, Apache Spark distributed framework has been taken as our reference in this paper which is designed for stream data processing and having support of Java, Scala and Python programming languages [14]. Apache Spark [10, 15] is a distributed programming framework which is designed for batch and stream (real time) data processing. Due to in-memory computation support, the processing speed of Apache Spark is much faster than disk based computation supported by Hadoop (On-Disk Computation Model which will be discussed in brief in the next section). Having support of Java, Python and Scala, the algorithms can be implemented.

In these programming languages; since, all these programming languages are having high level APIs, it supports the graph execution too two of these techniques will be described below:

Standalone: A simple cluster manager included with Spark that makes it easy to set up a cluster. It could be launched either manually, by starting a master and workers by hand, or use one of the provided launch scripts from the official website of apache spark. It is also possible to run these daemons on a single machine for testing [16].

Apache Mesos: The cluster manager in the below diagram is a Spark master instance. When using Mesos, the Mesos master replaces the Spark master as the cluster manager [17] (Fig. 1).

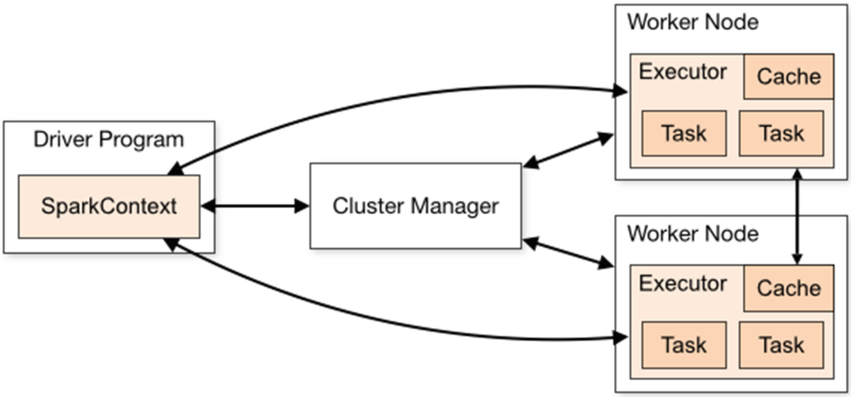


Fig. 1. Mesos cluster manager

Now when a driver creates a job and starts issuing tasks for scheduling, Mesos determines what machines handle what tasks. Because it takes into account other frameworks when scheduling these many short-lived tasks, multiple frameworks can coexist on the same cluster without resorting to a static partitioning of resources [17].

3 K-Means Example for Spark Machine Learning Library

K-Means is a popular clustering method. Clustering methods are used when there is no class to be predicted but instances are divided into groups or clusters. The clusters hopefully will represent some mechanism at play that draws the instance to a particular cluster. The instances assigned to the cluster should have a strong resemblance to each other. A typical use case for K-Means is segmentation of data. For example, suppose that there is a study for heart disease and there is a theory that individuals with heart disease are overweight [10].

The data that we are going to use in here is an example of Airplane trips in the United States of America having 1048576 attributes and 22 columns such as the year, month, day of month and so on. This data set is collected from years between 1987 to 2008. This data set contain a real data for the united states of america which describes the delay, distance and much more info for the airplane flights.

Below is a sample of the data set used in this thesis. We will use these attributes to pass into our K-Means algorithm. The below table describes the data set which will be used in this thesis (Table 1).

Table 1. Description of dataset

Name	Description
Year	1987-2008
Month	1-12
DayofMonth	1-31
DayOfWeek	1 (Monday) - 7 (Sunday)
DepTime	actual departure time (local, hhmm)
CRSDepTime	scheduled departure time (local, hhmm)
ArrTime	actual arrival time (local, hhmm)
CRSArrTime	scheduled arrival time (local, hhmm)
UniqueCarrier	unique carrier code
FlightNum	flight number
TailNum	plane tail number
ActualElapsedTime	in minutes
CRSElapsedTime	in minutes
AirTime	in minutes
ArrDelay	arrival delay, in minutes
DepDelay	departure delay, in minutes
Origin	origin IATA airport code
Dest	destination IATA airport code
Distance	in miles
TaxiIn	taxi in time, in minutes
TaxiOut	taxi out time in minutes
Cancelled	was the flight cancelled?

4 Proposed Model

To illustrate the sequence of how our proposed model works we will divide it into Three blocks to show how it will work (Fig. 2).

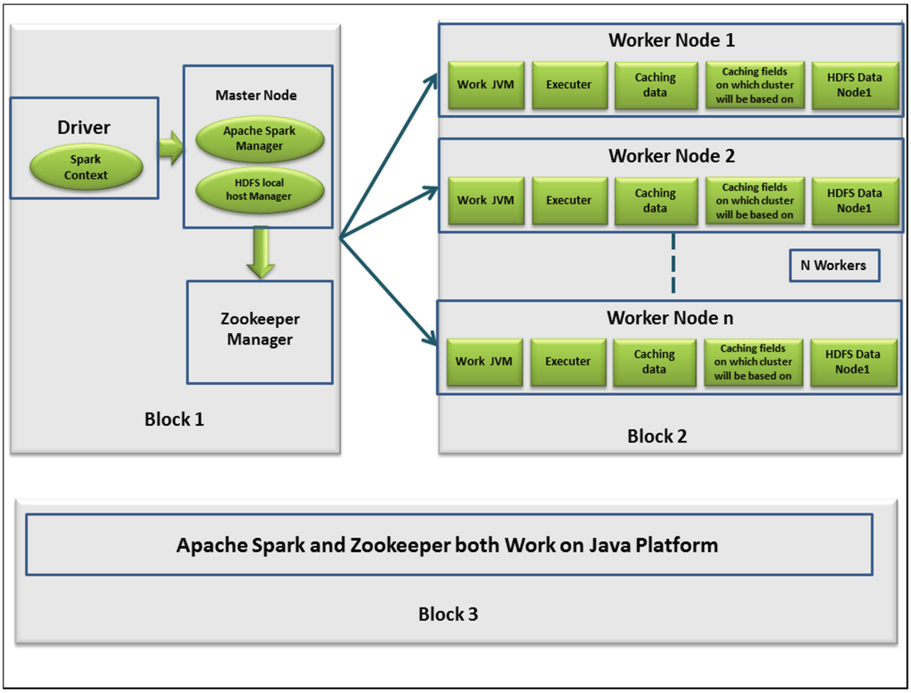


Fig. 2. Proposed model

4.1 Block 1

The data here will be inserted in the driver which holds the Apache spark then it will be passed to both the Apache spark manager and the Zookeeper -a tool used so that when the main cluster manager is down it works as a backup for it- and in the ground level we will have the Java platform of the virtual machine.

4.2 Block 2

Depending on the configuration text files where the number of the nodes, workers and cores are defined we will know the number of workers which we will work on. first the worker java platform will be initiated then a test connection between the node(s) and the manager is performed so that the executer starts then the data will be cached and distributed on the Hadoop Distributed file manager -as we are working on Apache spark over hadoop- then the result will appear and be returned to the apache Spark manager.

4.3 Block 3

To initiate any classification after configuring the number of clusters, nodes and Iteration java platform needs to be installed and configured on the ubuntu server machine. It is preferred to use java platform version 7 or later as starting from this version it is the most stable version to run Apache Spark on.

5 Experimental Environment

To start our experiment on apache spark to achieve the proposed model we have built a whole new environment from scratch so that we could start testing the used datasets and also apply the used algorithm on the datasets which we have used in our test environment. Our environment has been built over Microsoft windows 10 operating system using EliteBook8570W -VMware Workstation is a hosted hypervisor that runs on x64 versions of Windows and Linux operating systems – processor i7 second generation and RAM 20 GB, VMware Workstation version 12.5.7.

The environment has been built over our workstation on VMware Virtualization technology we first installed Ubuntu 15.10 and gave it the following specification for the Master Cluster Manager:

- RAM 8 GB
- Processor 2 core

Both Apache spark master and Apache spark node have the same hardware specifications and also have the same operating system but they may differ in the network configuration and configuration files of the apache spark tool.

Having the virtual machine setup using Ubuntu version 15.10 using VMware machine we will start downloading the requested tool that will be used in building our environment so that we could start running algorithms on it with several datasets, the tools which we have used to build our environment to run apache spark will be download and installed on the virtual machine in the following sequence:

- Install Java SDK 7
- Install Scala 2.10.4
- We then turn off the virtual machine and create a clone of it so that the clone will be the apache spark node which is linked to the main Virtual machine.
- Install SSH Remote Access on Apache spark node (clone).
- On the main machine (Apache Spark Master) we generate a RSA key for remote access.

6 Results and Analysis

Below diagrams will illustrate how results has been calculated by fixing the number of nodes (having two nodes), Iterations (having twenty iterations) and having variant number of clusters, experiments has been made on four different number of nodes the difference is that once it will be done without caching the parameters which the cluster will be base on and once with caching the data on which the classification will be based on.

According to Figs. 3, 4, 5 and 6 were group A is the group on which the data caching was not applied but on the other hand group B is the group on which caching the data is applied the difference in time could be noticed as follow in Fig. 3 were two clusters are applied the difference between group “A” and “B” is 0.2 s. Figure 4 were two clusters are applied the difference between group “A” and “B” is 2.2 s. Figure 5 were two clusters are applied the difference between group “A” and “B” is 0.1 s. Figure 6 were two clusters are applied the difference between group “A” and “B” is 0.3 s.

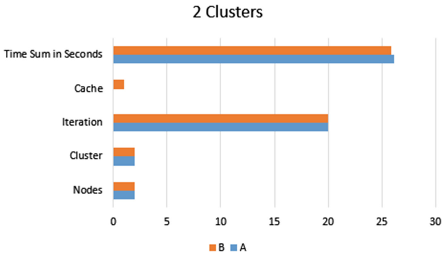


Fig. 3. Two clusters

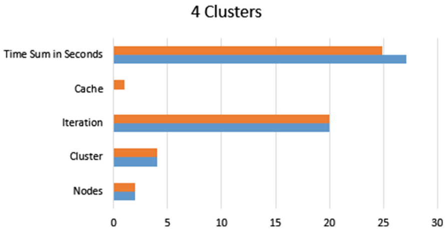


Fig. 4. Four clusters

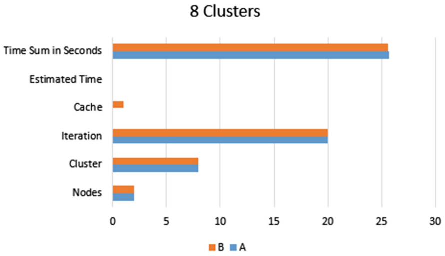


Fig. 5. Eight clusters

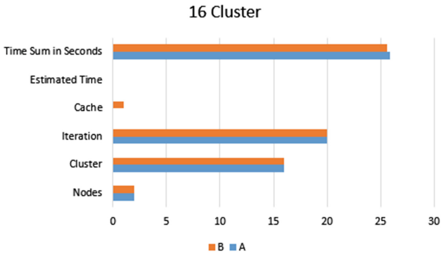


Fig. 6. Sixteen clusters

Below diagrams will illustrate how results have been calculated by fixing the number of nodes (having two nodes), Cluster (having two clusters) and having variant number of Iterations, experiments have been made on four different number of nodes the difference is that once it will be done without caching the parameters which the cluster will be base on and once with caching the data on which the classification will be based on.

According to Figs. 7, 8, 9 and 10 were group A is the group on which the data caching was not applied but on the other hand group B is the group on which caching the data is applied the difference in time could be noticed as follow in Fig. 7 were two clusters are applied the difference between group “A” and “B” is 0.5 s. Figure 8 were two clusters are applied the difference between group “A” and “B” is 0.2 s. Figure 9 were two clusters are applied the difference between group “A” and “B” is 1.2 s. Figure 10 were two clusters are applied the difference between group “A” and “B” is 1.5 s.

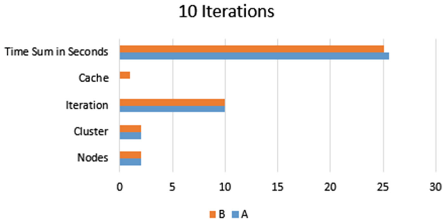


Fig. 7. Ten iterations

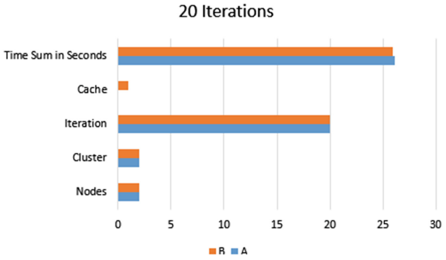


Fig. 8. Twenty iterations

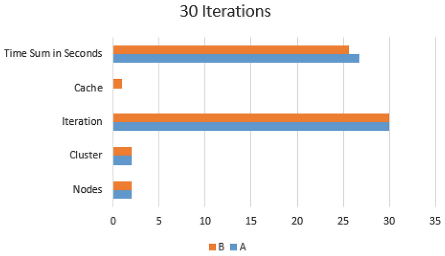


Fig. 9. Thirty iterations

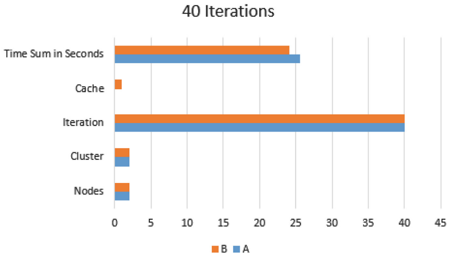


Fig. 10. Forty iterations

Below diagrams will illustrate how results have been calculated by fixing the number of Iterations (having twenty iterations), Cluster (having two clusters) and having variant number of Nodes, experiments have been made on four different number of nodes the difference is that once it will be done without caching the parameters which the cluster will be base on and once with caching the data on which the classification will be based on.

According to Figs. 11, 12, 13 and 14 were group A is the group on which the data caching was not applied but on the other hand group B is the group on which caching the data is applied the difference in time could be noticed as follow in Fig. 11 were two clusters are applied the difference between group “A” and “B” is 3.5 s. Figure 12 were two clusters are applied the difference between group “A” and “B” is 0.2 s. Figure 13 were two clusters are applied the difference between group “A” and “B” is 2.8 s.

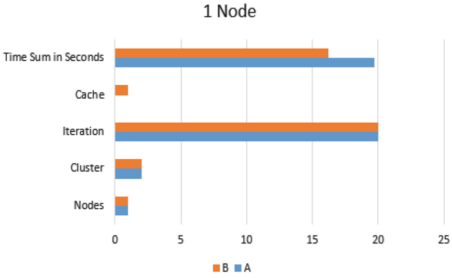


Fig. 11. One node

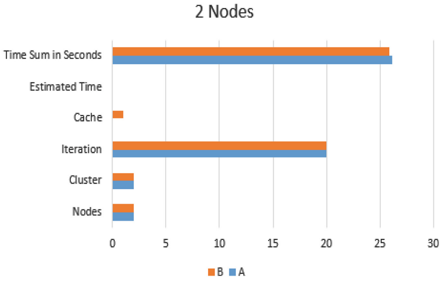


Fig. 12. Two nodes

Figure 14 were two clusters are applied the difference between group “A” and “B” is 0.827 s.

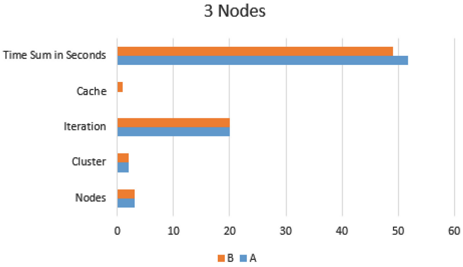


Fig. 13. Three node

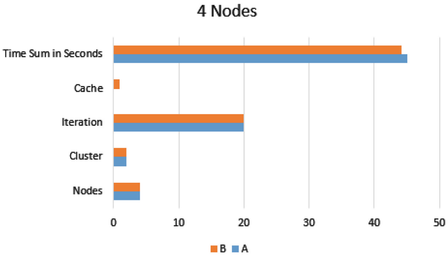


Fig. 14. Four nodes

7 Conclusion

This paper presented a performance enhancement framework for jobs running on Apache Spark platform. We have made a modification in the algorithm so that the time estimated for analyzing the data could be reduced as by default if we run the k-means algorithm without caching any data before the algorithm starts we will have a warning that we did not parse the used fields. But if we have noticed that if we parsed the fields on which the data will be classified on we will find that it will differ in the time estimated.

Comparison has been done on three different cases to measure the difference in time, the first case is having different nodes -working on four different cases- the time average when the fields which the classification is based on not passed before cluster start compared to that where the fields are passed before the average percentage of time saving is reduced to 3.40%. The second case is having different Clusters -working on four different cases- the time average when the fields which the classification is based on not passed before cluster start compared to that where the fields are passed before the average percentage of time saving is reduced to 2.57%. The third is having different Iterations -working on four different cases-the time average when the fields which the classification is based on not passed before cluster start compared to that where the fields are passed before the average percentage of time saving is reduced to 3.40%.

References

1. George, G., et al.: Big data and data science methods for management research. Acad. Manag. J. **59**, 1493–1507 (2016)
2. Ularu, E.G., et al.: Perspectives on big data and big data analytics. Database Syst. J. **3**, 3–14 (2012)
3. Assunção, M.D., et al.: Big data computing and clouds: trends and future directions. J. Parallel Distrib. Comput. **79**, 3–15 (2015)
4. Ferguson, A.G.: Big data and predictive reasonable suspicion (2014)
5. John Walker, S.: Big data: a revolution that will transform how we live, work, and think (2014)

6. Zikopoulos, P., Eaton, C.: Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data. McGraw-Hill Osborne Media, New York (2011)
7. Sharma, S., et al.: A brief review on leading big data models. Data Sci. J. **13**, 138–157 (2014)
8. Sharma, S., et al.: Leading NoSQL models for handling big data: a brief review. Int. J. Bus. Inf. Syst. **22**, 1–25 (2016)
9. Chen, H., et al.: Business intelligence and analytics: from big data to big impact. MIS Q. **36**, 1165–1188 (2012)
10. Meng, X., et al.: Mllib: machine learning in Apache Spark. J. Mach. Learn. Res. **17**, 1235–1241 (2016)
11. Khan, N., et al.: Big data: survey, technologies, opportunities, and challenges. Sci. World J. **2014**, 1–18 (2014)
12. Chen, C.P., Zhang, C.-Y.: Data-intensive applications, challenges, techniques and technologies: a survey on Big Data. Inf. Sci. **275**, 314–347 (2014)
13. Hafez, M.M., et al.: Effective selection of machine learning algorithms for big data analytics using Apache Spark. In: International Conference on Advanced Intelligent Systems and Informatics, pp. 692–704 (2016)
14. Armbrust, M., et al.: Scaling spark in the real world: performance and usability. Proc. VLDB Endow. **8**, 1840–1843 (2015)
15. Shoro, A.G., Soomro, T.R.: Big data analysis: Apache Spark perspective. Glob. J. Comput. Sci. Technol. **15** (2015)
16. Armbrust, M., et al.: Spark SQL: relational data processing in spark. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, pp. 1383–1394 (2015)
17. Saha, P., et al.: Integrating Apache Airavata with Docker, Marathon, and Mesos. Concurr. Comput. Pract. Exp. **28**, 1952–1959 (2016)

Open Circuit Fault Diagnosis of Cascaded H-Bridge MLI Using k-NN Classifier Based on PPCA

Nagendra Vara Prasad Kuraku, Murad Ali, and Yigang He(✉)

School of Electrical Engineering and Automation, Hefei University of Technology,
Hefei, People's Republic of China
Prasadkp123@gmail.com, 3176388451@qq.com, 18655136887@163.com

Abstract. Nowadays, great progress has been made in the development of multilevel inverters in renewable energy sources and other electrical drive applications. A k-Nearest Neighbors (k-NN) algorithm is applied to fault diagnosis of Cascaded H-Bridge Multilevel Inverter (CHMLI), this new fault diagnosis method is based on Probabilistic Principle Component Analysis (PPCA). The output voltage signals under different fault conditions of CHMLI are taken as the fault characteristics signals to avoid the effect of load variation on fault diagnosis. PPCA is used to optimize the data without changing the original properties of the input data, and k-NN is used to identify the accurate fault location and diagnosis the fault. The proposed technique is validated by conducting the experiment using Field-Programmable Gate Array (FPGA) controller. The simulation and experimental results shows that the proposed fault diagnosis method reduced the fault diagnosis time and improved the accuracy.

Keywords: 5-level MLI · Fault diagnosis · Fault features
Probabilistic principle component analysis and k-NN

1 Introduction

In recent years, industry has begun to demand higher power ratings, and multilevel inverter systems have become a solution for high power applications [1]. The multilevel inverter has particular advantages and is used in many applications such as pipeline pumps, rolling machines, and railway electrical traction drive systems.

The multilevel inverters having a high number of power semiconductors, and consequently, the possibility of a failure is much higher. Hence, the identification of possible faults and the operation under faulty conditions are of paramount importance. Due to the high number of components, the detection of a fault can be complicated in principle. Knowledge of fault prediction, fault behaviors, and fault diagnosis are necessary to maintain continuous operation of the multilevel inverter system. Some examples of these advanced methods and techniques based on frequency analysis [2], the use of neural networks (NNs) to search for some specific patterns, and the study of the time behavior in voltages and currents at the load [3]. There are many existing methods for the CHMLI have been proposed to identify and diagnosis the fault [4].

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 426–436, 2018.
https://doi.org/10.1007/978-3-319-74690-6_42

Various fault diagnosis techniques have been proposed for multilevel inverters for years. In [5], the different types of converter fault types have been investigated for voltage-fed inverter induction motor drive. A review of different fault detection methods for power inverters is presented in [6]. The fault detection in a 5-level diode-clamped multilevel inverter using wavelet analysis of output voltages and input DC currents is discussed in [7]. The different strategies for fault identification in multilevel inverters are discussed in [8].

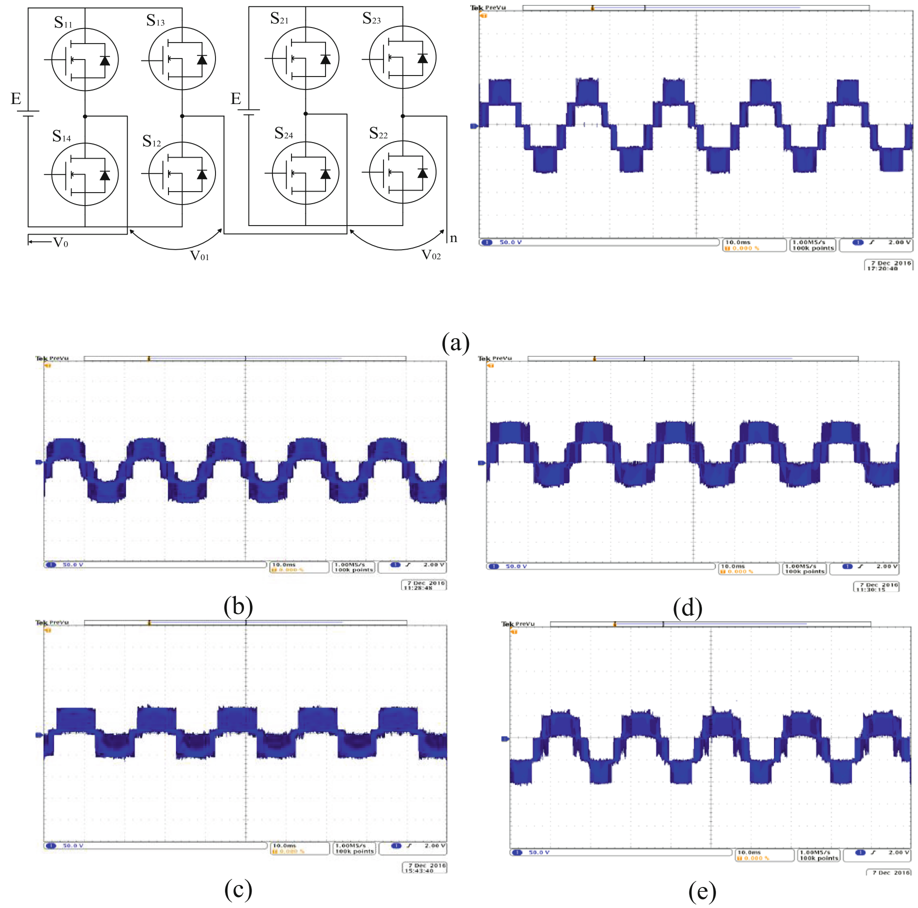
To overcome the problems identified in the exiting methods and to reduce the diagnosis computing time and improve the accuracy of fault diagnosis in CHMLI, this paper introduces a new method based on PPCA-k-NN algorithm. When the MOSFET's are open circuit state, the output voltage signal characteristics have been taken as the fault signals. The PPCA performs the voltage input signal transformation, with rated signal values as important features, and the output of the transformed signal is transferred to the k-NN classification. The k-NN is trained with both normal and abnormal data for the CHMLI: thus, the output of this network is nearly 0 and 1 as binary code. The binary code is sent to the fault diagnosis to decode the fault type and its fault location.

2 Open Circuit (OC) Fault Analysis of CHMLI

There are many possibilities to get faults in MOSFET module due to open or short circuit of antiparallel diode and open or short circuit of MOSFET. If OC fault occur due to semiconductor switch, then the current flowing through the switch to the load is disconnecting. The voltage across the switch will be zero and high current will flow through the switch to the load when SC fault occur due to the semiconductor switch. In this paper, mainly concentrated on OC faults due to semiconductor switch. In any inverter, there are different OC fault categories are present based on number of switching faults occur at a time. For n number of switches, n − 1 device faults are possible to get, those are single device fault, two device faults, n − 1 device faults and so on. Here we considered only single device faults as simple faults (Faults occur due to only anyone switch get open at a time) and two device faults called as complicated faults (fault occur due two switches get open at a time). Total 37 possibilities are there to get simple and complicated faults in 5-level CHMLI, here we considered all the 37 OC faults for CHMLI. Single phase cascaded H-bridge MLI simulation model is controlled by Phase Shift Pulse Width Modulation (PSPWM) technique. PSPWM is a simple and often used modulation technique for multilevel inverters [9]. In PSPWM technique for l-level inverter, l-1 carriers with the same frequency and the amplitude. A reference signal with the amplitude and the frequency has its zero cantered in the middle of the triangular carrier set.

The reference is continuously compared with each of the carrier signals. If the reference is greater than the carrier, the control signal is such that the corresponding output voltage level is high. Otherwise, if the reference is lower than the carrier, then the corresponding output voltage level is low.

The carrier signals, the sinusoidal reference signal with an amplitude modulation factor ma = 0.8. The inverter healthy state output voltage wave form with load resistance $R_L = 10\ \mathrm{k}\Omega$ is shown in Fig. 1(a).



(a)

(b)

(d)

(c)

(e)

Fig. 1. MLI output voltages under different fault conditions (a) fault free condition (b) OC fault occur at S11 (c) OC fault occur at {S13, S12} (d) OC fault occur at S13 (e) OC fault occur at S21

In n-level multilevel inverter, the amplitude modulation index m_a and the frequency modulation index, m_f, are defined as

$$m_a = \frac{V_r}{(n-1)V_c}, \tag{1}$$

$$m_f = \frac{f_c}{f_r} \tag{2}$$

the parameters considered are $n = 5$, $V_r = 1.6$ V, $V_c = 2$ V, $f_r = 50$ Hz, and $f_c = 6250$ Hz.

The output voltage signals of CHMLI under fault free condition and different fault conditions are shown in Fig. 1(a)–(e). In Fig. 1(b), the fault is occurred at switch S11, so it gets open circuit, then there is no positive supply to the load from the source through the switch. It can be observed in the output voltage waveform, it contains only one step in the positive half cycle, i.e. +50 V, but in negative half cycle there are two steps, those are −50 V and −100 V, because the switches in the second H-bridge are fault free. We can conclude from Fig. 1(b), the output voltage wave form get distortion, when any fault occurs in the CHMLI. The output current waveforms are independent from the OC faults, so the output voltage signals of CHMLI are taken as characteristic input signals to the classifier. To diagnosis these OC faults, in the present study PPCA-k-NN fault diagnosis method is used. PPCA is the feature extractor and k-NN is the feature classifier.

3 Proposed Fault Diagnosis Method

The proposed technique for a fault diagnostic system is illustrated in Fig. 2. Here the output voltage signals from the single phase 5-level cascaded MLI are taken as the fault signals, and given to the PPCA-k-NN fault diagnosis system. PPCA-k-NN fault diagnosis system mainly consisting of feature extraction (PPCA), Feature classifier (k-NN) and switching pattern calculation system. In this paper, mainly concentrated on feature extraction and feature classifier. The fault signals taken from CHMLI is consisting of very huge data, so the classifier takes more time to classify the fault location. PPCA feature extraction is used here to optimize the fault signals data. The optimized data from the PPCA is given to the k-NN, and it gives the output in the form of binary codes. After that, the binary codes are compared with prior knowledge to decode the fault type and its location. For the simple faults, the binary codes are shown in Table 1. Here ‘0’ represents the faulty condition and ‘1’ represents the healthy condition. If the OC fault occur at S_{21}, then the classification model output is $[1\ 1\ 1\ 1\ 1\ 0\ 1\ 1\ 1]^T$, and its output voltage signal is shown in Fig. 1(e).

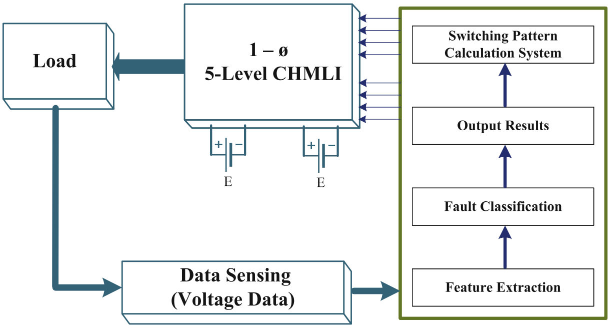


Fig. 2. Structure of fault diagnosis system

Table 1. Fault labels & class

Fault modes (Open circuit)	Labels	Fault class
Normal	[1 1 1 1 1 1 1 1 1]^{T}	1
S_{11}	[1 0 1 1 1 1 1 1 1]^{T}	2
S_{12}	[1 1 0 1 1 1 1 1 1]^{T}	3
S_{13}	[1 1 1 0 1 1 1 1 1]^{T}	4
S_{14}	[1 1 1 1 0 1 1 1 1]^{T}	5
S_{21}	[1 1 1 1 1 0 1 1 1]^{T}	6
S_{22}	[1 1 1 1 1 1 0 1 1]^{T}	7
S_{23}	[1 1 1 1 1 1 1 0 1]^{T}	8
S_{24}	[1 1 1 1 1 1 1 1 0]^{T}	9

3.1 Probabilistic PCA Based Feature Extraction

Here we used linear dimension reduction technique for which there exist computationally efficient technique, in this technique the high dimensional data point p^n is projected down to the lower dimensional vector x by

$$\mathrm{x} = \mathrm{F}p^n - \mathrm{constant}$$

Principal Component Analysis (PCA) is a well-established model for dimensionality reduction. Nevertheless, one limiting disadvantage of this technique is the absence of an associated probability density model or generative model [10]. PPCA overcome this problem. A latent variable model seeks to relate a d - dimensional observed data vector v:

$$v = y(x;U) + \varepsilon \tag{3}$$

Where x is q - dimensional vector of latent variable, ε is an x - independent noise process and U is the parameters, Eq. (3) induces a corresponding distribution in the data space & the model parameters may then be determined by maximum-likelihood (ML) techniques. Perhaps the most common example of a latent variable model is that of statistical factor analysis (Bartholomew 1987), in which the mapping is a linear function [11] of x:

$$v = Ux + \xi + \varepsilon \tag{4}$$

Latent variables are defined to be independent & Gaussian with unit variance, so $x \sim \mathfrak{R}(0, \mathrm{I})$. The noise model is also isotropic Gaussian such that $\varepsilon \sim \mathfrak{R}(0, \sigma^2\mathrm{I})$, with σ^2I diagonal, and $(d * q)$ parameter matrix U contains the factor loadings, the observation vectors are also normally distributed $v \sim \mathfrak{R}$ (ξ, $\mathbb{C}$), where the model covariance model is $\mathbb{C} = \sigma^2 I + UU^T$. The corresponding [12] log-likelihood is then, The ML estimator for ξ is given by the mean of the data, Estimates for U and σ^2 may be obtained [13] by iterative maximization of $\wp$. The conditional distribution of the latent variables x given the observed v, calculated using Bayes' rule and is again Gaussian:

$$\mathrm{P}(x|v) \sim \Re(M^{-1}U^{T}(v - \xi), \sigma^{2}M^{-1}) \tag{5}$$

Where we have defined $M = U^{T}U + \sigma^{2}I$. Note that M is of size $q * q$ while $\mathbb{C}$ is $d * d$.

$$U_{ML} = U_{q}\left(\Lambda_{q} - \sigma^{2}I_{q}\right)^{1/2}\aleph \tag{6}$$

Where the q column vectors in the $d * q$ matrix U_q are the principal eigenvectors of $\mathcal{H}$, the corresponding eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_q$ in the diagonal matrix Λ_q, and $\aleph$ is an arbitrary $q * q$ orthogonal rotation matrix Eq. (6).

$$\xi = \frac{1}{k}\sum\nolimits_{i=1}^{k} V_{i} \tag{7}$$

$$\sigma_{ML}^{2} = \frac{1}{d-q}\sum\nolimits_{i=q+1}^{d} \lambda_{i} \tag{8}$$

Note that in the PCA, one would take $U = U_q$, but this choice is not optimal in the sense of the ML for the PPCA model. The diagonal matrix $\left(\Lambda_{q} - \sigma^{2}I_{q}\right)^{1/2}$ gives an appropriate weight to each column vector of the matrix U_q. the conditional probability distribution of v given x

$$\mathrm{P}\,(^{v}/_{x}) = (2\pi\sigma^{2})^{-1/2}\exp\left(-\frac{1}{2\sigma^{2}}\|v - U - \bar{v}\|\right) \tag{9}$$

Where $U = U_{q}\left(\Lambda_{q} - \sigma^{2}I_{q}\right)^{1/2}x$. Hence, the ML reconstructed data point is taken as

$$\widehat{v} = U_{q}\left(\Lambda_{q} - \sigma^{2}I_{q}\right)^{1/2}x + \bar{v} \tag{10}$$

In that case, the reduction map is defined by

$$\hat{x} = \left(\Lambda_{q} - \sigma^{2}I_{q}\right)^{1/2}U_{q}^{t}(v - \bar{v}) \tag{11}$$

In order to minimize the average reconstruction error (optimal in the lease square sense)

$$\varepsilon = \frac{1}{n}\sum\nolimits_{i=1}^{n}\left\|v_{i} - U_{q}U_{q}^{T}\left(v_{i} - \bar{v}\right) - \bar{v}\right\|^{2} \tag{12}$$

These reconstruction and reduction maps were adopted in [14].

3.2 K-Nearest Neighbor (k-NN)

Now a day Many classifiers are used for fault detection and classification in inverters. In this paper k-NN is used for fault detection and fault classification in Cascaded

Multilevel Inverter. The k-NN algorithm has a good stability, high accuracy and is easy to implement. k-NN is a type of instance based learning, where the function is only approximated locally and all computation is deferred until classification [15]. The k-NN is the simplest algorithms of all machine learning algorithms.

In k-NN algorithm, the basic idea is to select ‘k’ samples with the minimum distance as nearest neighbours of the test data, and finally based on the categories of the k nearest neighbours to determine the distance of classification samples and each training sample. ‘k’ samples are randomly selected from training data, as the initial nearest neighbour samples.

For each test sample, the k-NN has to calculate all the known samples to obtain its k nearest neighbor points [16]. The training examples are vectors in a multidimensional feature space, each with a class label. The training phase of the algorithm consists only of storing the feature vectors and class labels of the training samples. If we assume the Euclidean distance as the dissimilarity measure, the k-NN algorithm considers a hypersphere centered on the test point *p*. We increase the radius r until the hypersphere contains exactly K points in the training data. The class label c(*p*) is then given by the most numerous class within the hypersphere. We need to choose the optimal ‘K’ value, if K value is very large, all the classifications will become the same and simply each novel *p* to the most numerous class in the training data. The best choice of k depends upon the data; generally, larger values of k reduce the effect of noise on the classification, but make boundaries between classes less distinct.

4 Discussion of Simulation and Experimental Results

This section deals with the simulation and experimental results of fault diagnosis of single phase cascaded 5-level multilevel inverter to validate the proposed method. To verify the effectiveness of the proposed diagnosis method, here simple and complicated faults are considered. The characteristics of output voltage signals of 5-level CHMLI under different simple and complicated faults are discussed in Sect. 2. The output voltage characteristic signals of 37 faults are taken at different values of modulation index (m_a), it is varied from 0.5 to 1 with an interval of 0.01. The simulation parameters are as fallows, frequency of the modulation index (m_f) is 125, sampling time (Ts) is 20 µs, sampling frequency (fs) is 50 kHz and simulation time is 18 ms. A 10% of Gaussian noise is added to the input data to test the proposed fault diagnosis technique performance.

The experiment has been done to validate the proposed PPCA-k-NN fault diagnosis method. The experimental setup is shown in Fig. 3. A 17N80C3 MOSFET module has been selected for the power devices in the CHMLI. The Darin source voltage (V_{DS}) is 800 V, continuous drain current (I_D) = 17A, and turn on delay time ($t_{d(on)}$) is 25 ns. The experimental parameters of the system are shown in Table 2. The driver circuit has consist of IR21844 integrated power modules. The experiment has been built based on XC3S250E FPGA to control the CHMLI using PSPWM technique.

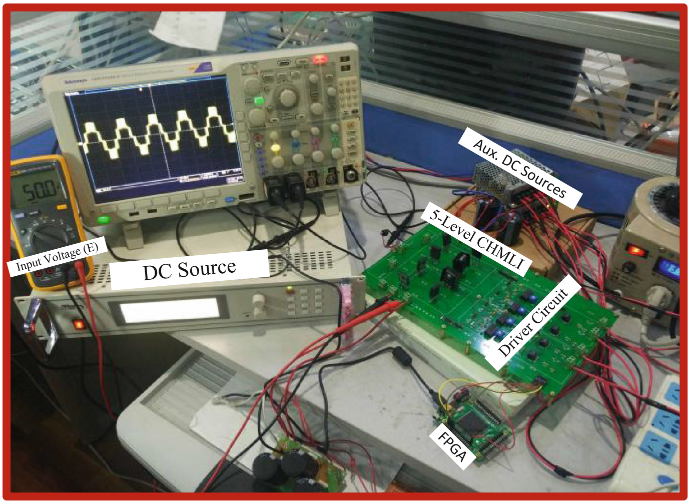


Fig. 3. Experimental setup

Table 2. System parameters

Variable	Description	Values
V_{dc}	DC-link voltage(V)(Sim.)	50
	DC-link voltage(V)(Exp.)	50
R_{load}	Sensitive load(kΩ)	10
f_r	Fundamental frequency(Hz)	50
f_c	Switching frequency(Hz)(Sim.)	6250
	Switching frequency(kHz)(Exp.)	25
m_a	Amplitude modulation index	0.8
T_s	Sampling time(μs)	20
T	Simulation time(ms)	18

Total Simulation Points (TSP) taken from the simulation is described by the below equation

$$TSP = 1 + \frac{T}{T_s} = 1 + \frac{18 \times 10^{-3}}{20 \times 10^{-6}} = 901 \tag{13}$$

Then the total output voltage signals data of different faults at different m_a values with TSP is represented in terms of [Trails × Dimensions], that is [1887 × 901]. After applied Fast Fourier Transform (FFT) transformation, the data dimensions are reduced from 901 to 513, and this data is given to the feature extractor.

The PC's of PCA contains only 95% of the total energy and it gives 93 PC's, but the PC's of PPCA contains 100% of the total energy. So, the feature information from the PC's of PPCA is more than from the PC's of PCA.

The main feature of the PPCA is, we can define the number of PC's in PPCA, which is not available in PCA. The number of dimensions present in the low dimensional projection data after feature extraction depends on number of PC's present in the feature extractor. The number of PC's used in PPCA are two. The PPCA gives low dimensional projection data compared to PCA, the PPCA gives reduced dimensional projection data of [1887 × 2], whereas the PCA gives [1887 × 93].

The optimized data from the PPCA is given to the k-NN classifier to classify the different faults. The k-NN classifier is tested for different models of k, those are Fine (k = 1), Medium (k = 10) and Coarse (k = 100). Upto the medium, k = 10 model, the accuracy of the k-NN classifier is high. The distance metric is Euclidean and distance weight is Equal are considered here for the k-NN classifier to choose the range of the k value. For all categories of different train data, the fault diagnosis accuracy of CHMLI with PPCA-k-NN method gives 100% accuracy.

After PPCA, the total sampling data given to the classifier is in five categories based on amount of train data, and the proposed PPCA-k-NN method is verified in each category at different distance weights of the k-NN classifier, also verified the results by using different distance metric's. In every category, the simulation done for 53 times and average parameters configuration is shown in Table 3.

Table 3. Simulation and experimental results

Category	Train data	Classifier: kNN		Simulation results				Experimental results	
				FFT-PCA-k-NN		PPCA-k-NN		PPCA-k-NN	
		Distance metric	Distance weight	Accuracy (%)	Time (sec)	Accuracy (%)	Time (sec)	Accuracy (%)	Time (sec)
I	95% (1793)	Euclidean	Equal	85.10	0.962	90.29	**0.867**	86.27	0.588
			L1	94.5	0.464	**100.00**	**0.345**	99.74	0.412
		Minkowski	Equal	83.80	1.837	89.84	**0.475**	85.34	0.587
			L1	97.08	1.259	**100.00**	**0.441**	**100.00**	0.561
II	85% (1604)	Euclidean	Equal	86.84	0.796	91.57	**0.470**	89.62	0.541
			L1	95.21	0.735	**100.00**	**0.390**	99.67	0.491
		Minkowski	Equal	82.12	1.744	91.88	**0.451**	90.05	0.535
			L1	97.96	1.721	**100.00**	**0.403**	98.93	0.558
III	70% (1321)	Euclidean	Equal	75.64	1.491	82.91	**0.476**	80.24	0.554
			L1	96.69	0.851	**100.00**	**0.396**	97.88	0.412
		Minkowski	Equal	70.27	5.709	71.00	**0.441**	65.28	0.337
			L1	97.84	4.742	**100.00**	**0.417**	98.64	0.534

In Table 3 shows the simulation and experimental results. In the first category (95% Train Data) the average diagnostic validation accuracy of FFT-PCA-k-NN using Euclidean distance matric for different distance weights is about 85.10% to 94.5 % and the running time is about 0.464 to 0.962 s. In the same category the PPCA-k-NN gives, the average diagnostic validation accuracy is reaches to 90.29% to 100% and even the time taken to diagnosis process is also very low i.e., from 0.345 to 0.867 s. at all different train data, the proposed PPCA-k-NN gives better diagnosis accuracy and less running time. In the k-NN classifier, the test data distances to the 'k' samples are calculated by using different distance matric.

In all distance matric, the 'Euclidean Distance' matric is best suitable for the k-NN classifier in fault diagnosis of CHMLI, because it gives high accuracy and less running time at Inverse distance weight for all categories of different train data. Based on simulation and experimental results shown in Table 3, the proposed PPCA-k-NN method is improved the diagnosis accuracy to its maximum value, and also reduces the time taken for diagnosis process under different categories of train data. Also can conclude that the proposed PPCA-k-NN method gives the maximum accuracy of 100% with less running time of 0.345 s in first category of 95% train data with Euclidean distance matric at Inverse distance weight.

5 Conclusion

A new fault diagnosis strategy using PPCA based k-NN for CHMLI is presented. The approach has been evaluated and validated on experimental data issued from a CHMLI controlled by using FPGA. Based on the knowledge of the inverter behaviors, its output voltage signals have been selected as fault characteristic ones for the fault diagnosis strategy. In this paper PPCA feature extraction is introduced for better dimensional reduction of the fault feature data. The proposed method gives accurate and fast diagnosis for not only simple faults but also for complicated faults. The experimental and simulation results conclude that the proposed PPCA-k-NN method not only gives accurate fault location, but also do the fast diagnosis compared to the PCA-k-NN.

Acknowledgements. This work was supported by the National Natural Science Foundation of China under Grant No. 51577046, The State Key Program of National Natural Science Foundation of China under Grant No. 51637004, The national key research and development plan "Important Scientific Instruments and Equipment Development" of China Under Grant No. 2016YFF0102200.

References

1. Mariethoz, S.: Systematic design of high-performance hybrid cascaded multilevel inverters with active voltage balance and minimum switching losses. IEEE Trans. Power Electron. **28**, 3100–3113 (2013)
2. Khomfoi, S., Tolbert, L.M.: Fault diagnosis and reconfiguration for multilevel inverter drive using AI-based techniques. IEEE Trans. Ind. Electron. **54**, 2954–2968 (2007)
3. Khomfoi, S., Tolbert, L.M.: Fault diagnostic system for a multilevel inverter using a neural network. IEEE Trans. Power Electron. **22**, 1062–1069 (2007)
4. Wang, T., Xu, H., Han, J., Elbouchikhi, E., Benbouzid, M.E.H.: Cascaded H-bridge multilevel inverter system fault diagnosis using a PCA and multiclass relevance vector machine approach. IEEE Trans. Power Electron. **30**, 7006–7018 (2015)
5. Lu, B., Sharma, S.K.: A literature review of IGBT fault diagnostic and protection methods for power inverters. IEEE Trans. Ind. Appl. **45**(5), 1770–1777 (2009)
6. Keswani, R.A., Suryawanshi, H.M., Ballal, M.S.: Multi-resolution analysis for converter switch faults identification. IET Power Electron. **8**(5), 783–792 (2015)

7. Keswani, R.A., Suryawanshi, H.M., Ballal, M.S., Renge, M.M.: Wavelet modulus maxima for single switch open fault in multi-level inverter. Electric Power Compon. Syst. **42**(9), 889–900 (2014)
8. Lezana, P., Pou, J., Meynard, T.A., Rodriguez, J., Ceballos, S., Richardeau, F.: Survey on fault operation on multilevel inverters. IEEE Trans. Ind. Electron. **57**(7), 2207–2218 (2010)
9. Palanivel, P., Dash, S.S.: Analysis of THD and output voltage performance for cascaded multilevel inverter using carrier pulse width modulation techniques. IET Power Electron. **4**, 951–958 (2011)
10. Tipping, M.E., Bishop, C.M.: Mixtures of Probabilistic Principle Component Analysers, pp. 443–482. MIT Press, Cambridge (2006)
11. Tipping, M.E., Bishop, C.M.: Probabilistic principal component analysis. J. R. Stat. Soc. **61**(3), 611–622 (1999)
12. Benameur, S., Mignotte, M., Destrempes, F., Guise, J.A.D.: Three-dimensional biplanar reconstruction of scoliotic rib cage using the estimation of a mixture of probabilistic prior models. IEEE Trans. Biomed. Eng. **52**, 1713–1728 (2005)
13. Jon, E., Kim, D.K., Kim, N.S.: Robust correlation estimation for EMAP-based speaker adaptation. IEEE Sig. Process. Lett. **8**, 184–186 (2001)
14. Kim, D.K., Kim, N.S.: Rapid speaker adaptation using probabilistic principal component analysis. IEEE Sig. Process. Lett. **8**, 180–183 (2001)
15. He, Q.P., Wang, J.: Fault detection using the k-Nearest neighbor rule for semiconductor manufacturing processes. IEEE Trans. Semicond. Manuf. **20**(4), 345–354 (2007)
16. Zhou, Z., Wen, C., Yang, C.: Fault detection using random projections and k-Nearest neighbor rule for semiconductor manufacturing processes. IEEE Trans. Semicond. Manuf. **28**(1), 70–79 (2015)

Improved Production Key Performance Indicators (KPI's) Using Intelligent-Manufacturing Execution Systems (I-MES)

Mohamed I. Mahmoud[1](✉), Hossam Hassan Ammar[2],
Mostafa Hassan Eissa[3], and Muhammad M. Hamdy[4]

[1] Department of Industrial Electronics and Control Engineering,
Faculty of Electronic Engineering at Menouf, Menofia University,
Shibin El Kom, Egypt
m.i.mahmoud@el-eng.menofia.edu.eg

[2] School of Engineering and Applied Science, Nile University, Giza, Egypt
hhassan@nu.edu.eg

[3] Department of Physics and Mathematics Engineering,
Faculty of Electronic Engineering at Menouf, Menofia University,
Shibin El Kom, Egypt
meissa1946@yahoo.com

[4] Faculty of Engineering, Misr University for Science and Technology,
Giza, Egypt
drmuhhamdy@yahoo.com

Abstract. The aim of this research is to reduce the gap between manufacture expertise and management expertise by using modern technology like Manufacturing Execution System (MES) via Artificial Intelligence (AI) and Machin Learning (ML). A design of MES has been proposed and implemented on El-Araby Plastic Injection Molding (PIM) factory. This work is based on the International Society of Automation Standard (ISA-S95). A fully automated data management system has been designed and implemented to control data follow between shop floor e.g. (machines and operators) and management floor e.g. (production, quality, inventory and Enterprise Resource Planning (ERP) staff). A real-time MES quality control and monitoring has been also designed and implemented using either classic computing, or AI and ML techniques. Fuzzy Logic (FL) controllers have been designed and implemented as feedforward controllers; to improve the performance of existed classical PID controller of injection parameters. An expert FL system has been used as one of AI techniques to implement manufacturer expertise in MES. An FL product quality classifier as ML has been designed and implemented depending on injection molding conditions to give the expected product quality. An expert system has been devolved based on machine manufacturers, raw material suppliers and production engineer expertise's using FL to give the injection parameters set points according to the product quality measure. The final results of this work are an intelligent computing system named (I-MES).

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 437–448, 2018.
https://doi.org/10.1007/978-3-319-74690-6_43

Keywords: Production Key Performance (KPI) · Intelligent-Manufacturing Execution Systems (I-MES) · International Society of Automation Standard (ISA-S95) · Fuzzy Logic (FL) · Artificial Intelligence (AI) · Machine Learning (ML) · Quality Control (QC) · Enterprise Resource Planning (ERP) · Plastic Injection Molding (PIM)

1 Introduction

The competitiveness pressure today makes full use of all optimization opportunities for manufacturing processes as industrial actuators. These optimizations started with planning and engineering, continuing into operation and right maintenance, through expansion and modernization. The integrated automation requirements are: efficient configuration, faster integration and commissioning, greater flexibility in production and higher availability and energy saving. Since early 2000, ISA-S95 defines terminology and models that are used for integrating MES at the business level with automation systems at the production level [1]. The standard was declared as national norm by the American National Standard Institute (ANSI) [2]. MES are computerized systems used in manufacturing to provide the right information at the right time. MES also, show the manufacturing decision maker "how the current conditions on the plant floor can be optimized to improve productivity [3]. MES work in real time to enable control of multiple elements of the production process such as inputs, personnel, machines and support services [4]. MES is strongly connected to the manufacturing processes. MES is a rather new software system, which is linking the business systems in companies with the control systems on the factory shop floor [5]. Overall Equipment's Efficiency (OEE) is considered as the best tool to measure in real time KPI's such as the use of materials, the productivity according to batches, the machine breakdown [6]. Thus, it is possible to create dashboard by evaluating indicators such as the OEE, and the mean time between failures or the mean time to repair [7].

Nowadays, AI and ML have become smart solutions of most of complicated industry where conventional techniques disable to solve [9]. There are a lot of modern techniques of AI and ML to produce modeling and control industry problems [10]. An FL has been selected as AI technique due to its simplicity and efficiency of modeling and control of the industry problem [11]. FL controller is used to improve conventional controller performance due to industrial systems nonlinearity and uncertainty [12]. FL classifier has been designed and implemented from statistical analysis and features extraction of product quality based on injection parameters [13]. An expert system based on FL has been developed and implemented for injection molding products. MES expert system assists the machine operator to set the right value of the process variables to obtain high quality parts [14]. The expert system integrates data from the operator's experience, theoretical knowledge, material and machine recommendations, and experimental data from tests, to calculate the membership functions and rule base of fuzzy system [15]. This paper introduces implementation and developing an Intelligent MES (I-MES) to improve the production process performance KPI's in real time.

2 Quality Control System Based on Classical Computational

Investigations and analysis of the pervious quality control system have been done. It was found that it depends only on quality supervisor. It also gives poor products quality. The quality supervisor takes a random sample each one hour and not real-time quality monitoring. Quality reports are done manually using paper work information system. Analysis process is done at the day end as monthly quality reports. The concept of operation of this stage is to design a computerized and real-time quality inspection and quality control system. To supervise and control the quality of production process and gives accurate reports about it. Machine operator has been trained to be self-inspection of his products. It is a good idea but after enforcement, it was found that without supervision it is not effective, because of the machine operator wants to increase his productivity. Then he may changes the injection parameters to decrease production cycle time to increase his productivity.

2.1 Effect of Injection Molding Parameters on Product Quality

It must be a residual quantity of molded plastic at the end of injection process it is called cushion. Cushion definition: The melt material in barrel lifted after complete injection process e.g. (the hold time end) or at the end of (pack and hold) injection stage timer end [16]. Minimum Cushion Calculation: It must be a cushion in the barrel at least minimum cushion at the injection process to prevent the fraction between injection stroke and the injection nozzle.

Cushion is the measure of stroke position at the end of the hold time it depends on machine features and mold volume and shot size [17].

$$\text{Shot size } = \frac{W * 10}{A * \rho * \mu} + \text{min cushion}$$

W: product weight (g), A: sectional area of stroke (cm^2)

ρ: Specific weight of raw material (g/cm^2), μ: injection machine efficiency.

Experimental Study of Cushion Error and charging time on Product Quality: Apply trial and error method to study the effect of cushion error on product quality at Toshiba injection molding machine in El-Araby plastic injection molding factory, as shown in Tables 1 and 2.

Table 1. Experimental study of cushion error

Trail	Cushion mm	Test result	Trail	Cushion mm	Test result
1	8	Accepted	13	8	Accepted
2	8.5	Accepted	14	7.5	Accepted
3	9	Accepted	15	7	Accepted
4	9.5	Accepted	16	6.5	Accepted
5	10	Accepted	17	6	Accepted
6	10.5	Small short shot	18	5.5	Small flash
7	11	Small short shot	19	5	Small flash
8	11.5	Medium short shot	20	4.5	Medium flash
9	12	Medium short shot	21	4	Medium flash
10	12.5	Large short shot	22	3.5	Large flash
11	13	Large short shot	23	3	Large flash

Table 2. Experimental study of charging time

Trail	Char.-time sec	Test result	Trail	Char.-time sec	Test result
1	7	Accepted	13	7	Accepted
2	7.5	Accepted	14	6.5	Accepted
3	8	Accepted	15	6	Accepted
4	8.5	Accepted	16	5.5	Accepted
5	9	Accepted	17	5	Accepted
6	9.5	Small short shot	18	4.5	Small flash
7	10	Small short shot	19	4	Small flash
8	10.5	Medium short shot	20	3.5	Medium flash
9	11	Medium short shot	21	3	Medium flash
10	11.5	Large short shot	22	2.5	Large flash
11	12	Large short shot	23	2	Large flash

2.2 Pressure and Temperature Effect on Product Quality

The effects of injection temperature and pressure on plastic products quality have been investigated and results are plotted as shown in Fig. 1. It was founded that as the lower the temperature is, the higher pressure is needed to deliver the polymer melt into the cavity. If the temperature is too high, it is a risk causing material degradation. If the injection pressure is too low, a short shot may result. If the pressure is too high, flash product may result.

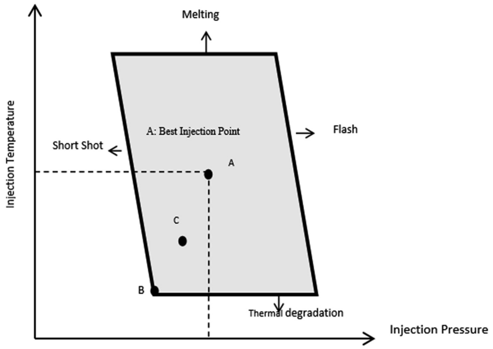
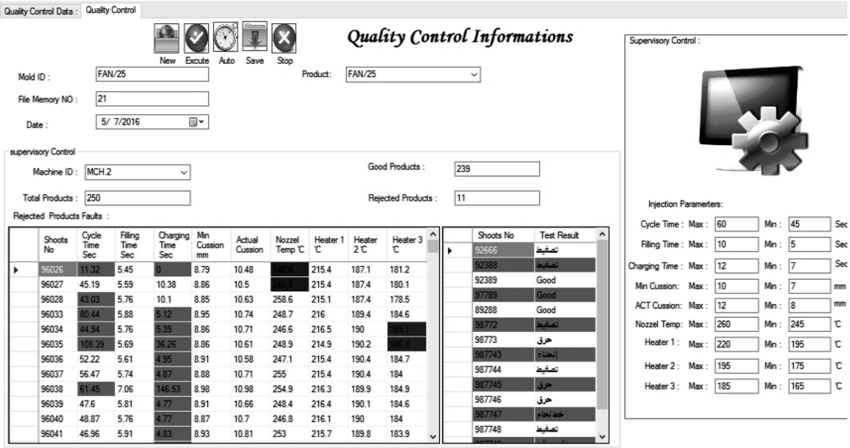


Fig. 1. Relation between PH and product quality

The effectiveness of each injection parameter on product quality has been studied. A trial and error method was applied to get the tolerance limits for each parameter and its effect on product quality. A decision-making program depending on the error of injection parameters has been designed and implemented as shown in Fig. 2. It also gives real time instructions to machine operator to overcome the defects by adjusting injection parameters based on conventional PID controller to be at point C which is the optimal point for quality and power saving. Point A is the best quality point but higher power consumption points and point C is lowest power consumption but poor quality.



Quality Control Informations

Mold ID : FAN/25

Product: FAN/25

File Memory NO : 21

Date : 5/ 7/2016

supervisory Control

Machine ID : MCH.2

Good Products : 239

Total Products : 250

Rejected Products : 11

Rejected Products Faults :

	Shoots No	Cycle Time Sec	Filling Time Sec	Charging Time Sec	Min Cussion mm	Actual Cussion	Nozzel Temp ℃	Heater 1 ℃	Heater 2 ℃	Heater 3 ℃
▸	96026	11.32	5.45	0	8.79	10.48	240.6	215.4	187.1	181.2
	96027	45.19	5.59	10.38	8.86	10.5	240.8	215.4	187.4	180.1
	96028	43.03	5.76	10.1	8.85	10.63	258.6	215.1	187.4	178.5
	96033	80.44	5.88	5.12	8.95	10.74	248.7	216	189.4	184.6
	96034	44.94	5.76	5.39	8.86	10.71	246.6	216.5	190	185.1
	96035	108.39	5.69	36.26	8.86	10.61	248.9	214.9	190.2	186.4
	96036	52.22	5.61	4.95	8.91	10.58	247.1	215.4	190.4	184.7
	96037	56.47	5.74	4.87	8.88	10.71	255	215.4	190.4	184
	96038	61.45	7.06	146.53	8.98	10.98	254.9	216.3	189.9	184.9
	96039	47.6	5.81	4.77	8.91	10.66	248.4	216.4	190.1	184.6
	96040	48.87	5.76	4.77	8.87	10.7	246.8	216.1	190	184
	96041	46.96	5.91	4.83	8.93	10.81	253	215.7	189.8	183.9

	Shoots No	Test Result
▸	92666	تصفيط
	92388	تصفيط
	92389	Good
	97789	Good
	89288	Good
	98772	تصفيط
	98773	حرق
	987743	إنحناء
	987744	تصفيط
	987745	حرق
	987746	حرق
	987747	خط لحام
	987748	تصفيط

Supervisory Control :

Injection Paramerters:

Cycle Time :	Max : 60	Min : 45	Sec
Filling Time :	Max : 10	Min : 5	Sec
Charging Time :	Max : 12	Min : 7	Sec
Min Cussion:	Max : 10	Min : 7	mm
ACT Cussion:	Max : 12	Min : 8	mm
Nozzel Temp:	Max : 260	Min : 245	℃
Heater 1 :	Max : 220	Min : 195	℃
Heater 2 :	Max : 195	Min : 175	℃
Heater 3 :	Max : 185	Min : 165	℃

Fig. 2. Real time injection parameters used for quality monitoring and control.

3 Quality Control Systems Based on Artificial Intelligence (AI) and Machine Learning (ML)

Fuzzy logic has been applied as one of machine intelligence techniques to implement manufacturer expertise in MES. First, we begin to develop Fuzzy Logic controller in order to overcome the cushion size error and improve conventional PID performance and another algorithm for Charging Time Error (CTE) to control product shot size. Second, a Fuzzy Logic products quality classifier has been designed and implemented depending on injection molding parameters to give the expected product quality. Third, a Fuzzy Logic expert system has been designed to give the injection parameters set points according to the product quality based on machine manufacturers and raw material supplier and production expertise as shown in Fig. 3.

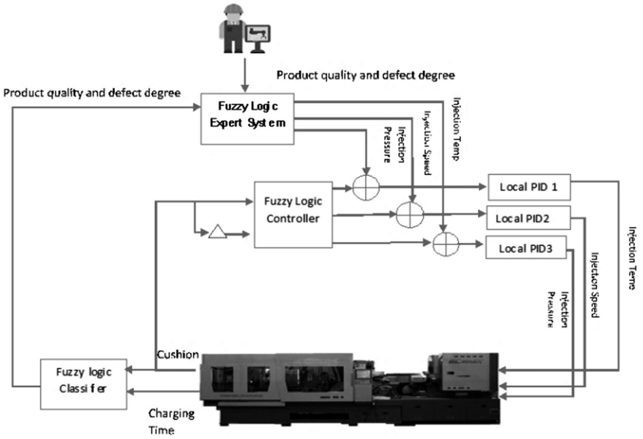


Fig. 3. Real time intelligent MES (I-MES) block diagram

4 Artificial Intelligence Controller for Product Shot Size

A fuzzy logic controller algorithm for cushion size error in order to control product shot size to improve conventional PID controller performance is described in Fig. 4.

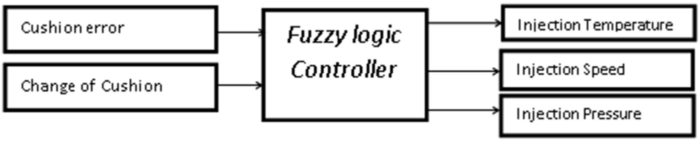
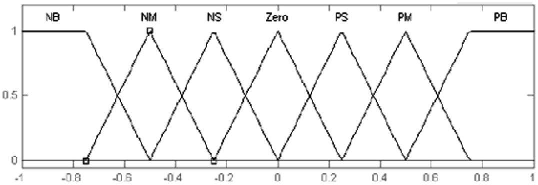


Fig. 4. Product shot size controller block diagram

4.1 Fuzzification of Input and Output

Two input parameters Error (E) and Change of Error (CE) in cushion and three outputs: injection speed, temperature and pressure were selected from the feature extraction process as shown in Fig. 5.



Linguistic Value	Range
Negative Large (N.L)	-100%: -50%
Negative Medium (N.M)	-75%: -25%
Negative Small (N.S)	-50%: -0%
Zero	-25%: +25%
Positive Small (P.S)	00%: +25%
Positive Medium (P.M)	+25%: +75%
Positive large (P.L)	50%: 100%

Fig. 5. Input and output membership functions

4.2 Controller Rules Tables, Rule Base and Defuzzification

Forty-nine rules were modelled and inferred from the domain expert for human control of the plastic product quality [20] as shown in Tables [3, 4 and 5]. Canter of Maxima (CoM) method proved to yield better results during the tuning stage of the fuzzy controller.

Table 3. Rules table of injection speed

E	EC						
	NB	NM	NS	ZO	PS	PM	PB
NB	PB	PB	PM	PM	PS	ZO	ZO
NM	PB	PB	PM	PS	PS	ZO	NS
NS	PM	PM	PM	PS	ZO	NS	NS
ZO	PM	PM	PS	ZO	NS	NM	NM
PS	PS	PS	ZO	NS	NS	NM	NM
PM	PS	ZO	NS	NM	NM	NM	NB
PB	ZO	ZO	NM	NM	NM	NB	NB

Table 4. Rules table of injection temp.

E	EC						
	NB	NM	NS	ZO	PS	PM	PB
NB	PS	NS	NB	NB	NB	NM	PS
NM	PS	NS	NB	NM	NM	NS	ZO
NS	ZO	NS	NM	NM	NS	NS	ZO
ZO	ZO	NS	NS	NS	NS	NS	ZO
PS	ZO	ZO	ZO	ZO	ZO	ZO	ZO
PM	PB	NS	PS	PS	PS	PS	PB
PB	PB	PM	PM	PM	PS	PS	PB

Table 5. Rules table of injection speed

E	EC						
	NB	NM	NS	ZO	PS	PM	PB
NB	NB	NB	NM	NM	NS	ZO	ZO
NM	NB	NB	NM	NS	NS	ZO	ZO
NS	NB	NM	NS	NS	ZO	PS	PS
ZO	NM	NM	NS	ZO	PS	PM	PM
PS	NM	NS	ZO	PS	PS	PM	PB
PM	ZO	ZO	PS	PS	PM	PB	PB
PB	ZO	ZO	PS	PM	PM	PB	PB

5 Machine Learning (ML) Using Fuzzy Logic Classifier for Products Quality

Design product quality classifier from measuring the cushion and charging time errors during injection molding process as shown in Fig. 6. The MES quality reports gives that the most common defected products are flash and short shot. The cushion and charging time are selected based on experimental and analysis process of the most common defects according to the mold and raw material manufactures related to El-Araby factory.

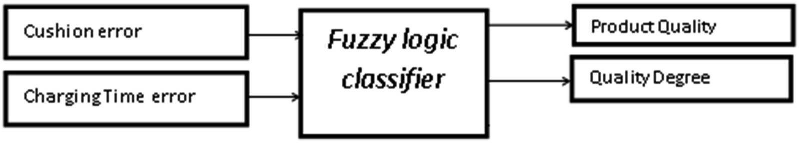
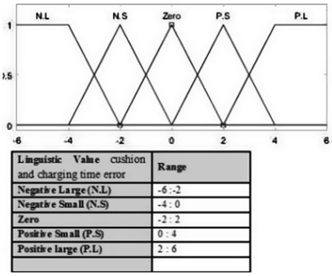


Fig. 6. Product quality classifier block diagram.

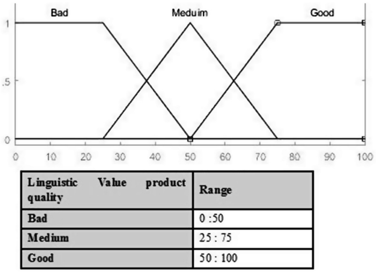
5.1 Fuzzification of Input and Output

Two input parameters were selected to the fuzzifier, namely cushion error and charging time error obtained from the feature extraction process shown in Fig. 7. The output variables were taken from the common problems encountered in plastic product quality inspection which are flash product and short shot, as shown in Fig. 8. The centres of output memberships are selected based on experimental trails and Mosaic or Table Lookup Scheme [19].



Linguistic Value cushion and charging time error	Range
Negative Large (N.L)	-6 :-2
Negative Small (N.S)	-4 : 0
Zero	-2 : 2
Positive Small (P.S)	0 : 4
Positive large (P.L)	2 : 6

Fig. 7. Input membership functions



Linguistic Value product quality	Range
Bad	0 :50
Medium	25 : 75
Good	50 : 100

Fig. 8. Output membership functions

5.2 Rules Table for Classification

Twenty-five rules were modelled and inferred from the domain expert for human classification of the plastic product quality [20] as shown in Table 6. The smallest of the maximum (SoM) method proved to yield better results during the tuning stage of the fuzzy classifier.

Table 6. Rules table of fuzzy classifier

Cushion	Ch_time error				
	N.L	N.S	Zero	P.S	P.L
N.L	L.SH	L.SH/B.ACC	L.SH/B.ACC	M.FS/M.ACC	S.FS/M.ACC
N.S	L.SH/B.ACC	M.SH/B.ACC	S.SH/M.ACC	S.FS/M.ACC	M.FS/M.ACC
Zero	L.SH/B.ACC	S.SH/M.ACC	G.ACC	S.FS/M.ACC	L.FS/B.ACC
P.S	M.SH/M.ACC	S.SH/M.ACC	S.FS/M.ACC	L.FS/B.ACC	L.FS/B.ACC
P.L	S.SH/M.ACC	S.SH/G.ACC	L.FS/B.ACC	L.FS/B.ACC	L.FS

SH: Short Shot. FS: Flash. ACC: Accepted.

6 Expert System (ES) to Defect Overcome Fuzzy Logic (FL)

The Proposed Fuzzy logic expert system takes the defect type and its degree and gives the set point of injection parameters as shown in Fig. 9. Table 7 collect the expertise of manufacturing and raw material supplier in defect overcome rules and their priority in order not causing conflict between defects.

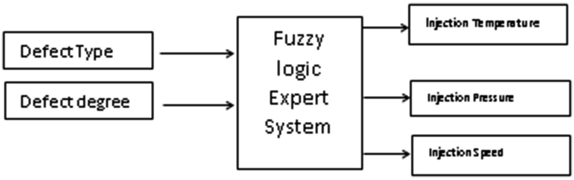
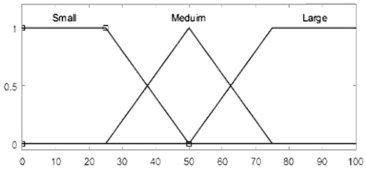


Fig. 9. Fuzzy logic expert system block diagram

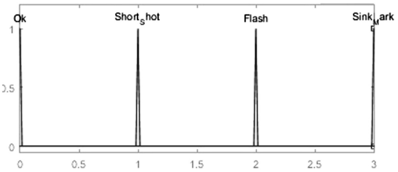
6.1 Fuzzification of Input and Output

Two input parameters were selected to be the input of the controller, namely Defect type and Defect degree. shown in Figs. [10, 11]. The proposed outputs are melt temperature, injection pressure and injection speed based on experimental and producer's expertise as shown in Fig. 12.



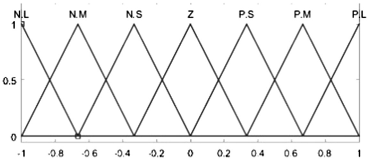
Linguistic Value	Range
Small	0 :50
Medium	25 : 75
Large	50 : 100

Fig. 10. Defect degree membership functions



Linguistic Value	Range
OK	0
Short Shot	1
Flash	2
Sink mark	3

Fig. 11. Defect type membership functions



Linguistic Value	Range
Negative Large (N.L)	-1:-0.75
Negative Medium (N.M)	-1:-0.25
Negative Small (N.S)	-0.75:0
Zero (Z)	-0.25:0.25
Positive Small (P.S)	0 : 0.75
Positive Medium (P.M)	0.25:1
Positive large (P.L)	0.75 : 1

Fig. 12. Output membership functions.

Table 7. Rules for fuzzy expert system

Parameter / Defect	Melt temperature	Injection speed	Injection pressure
Sink Marks	▨ 1	▨ 3	■ 2
Short Shot	■ 2	■ 1	■ 3
Flash	▨ 3	▨ 1	▨ 2
■	Increase		
▨	Decrease		

6.2 Rules Tables, Rule Base and Defuzzification for Fuzzy Expert System

Ten rules were modelled and inferred from the domain expert for human control of the plastic product quality as shown in Table 8. These rules are extracted from experimental work based on raw-material supplier and machine manufacturing and production expertise. The Gradient Descent method [19] is used to adjust center and type of input and output membership functions. Rules are arranged as shown in Table 8. The AND operation in applying rules was carried out using the product technique, implication was applied based on the smallest of the maximum (SoM) method.

Table 8. Rules for fuzzy expert system

Defect degree	Defect type			
	OK	Short shot	Flash	Sink mark
Small	Z. Injection press Z. Injection temp Z. Injection speed	P.S. Inj press Z. Inj temp Z. Inj speed	Z. Inj press Z. Inj temp P.S. Inj speed	Z. Inj press N.S. Inj temp Z. Inj speed
Medium		P.M. Inj press P.S. Inj temp Z. Inj speed	N.S. Inj press Z. Inj temp N.M. Inj speed	P.S. Inj press N.M. Inj temp Z. Inj speed
Large		P.L. Inj press P.M. Inj temp P.S. Inj speed	N.M. Inj press N.S. Inj temp N.L. Inj speed	P.M. Inj press N.L. Inj temp N.S. Inj speed

7 Results and Conclusion

Conclusions and future directions for further investigations are reported. All designed programs have been collected in one package and implemented at Toshiba El-Araby plastic injection molding factory step by step. The system has been tested and verified by Toshiba El-Araby engineers and staff [21]. Applying I-MES in El-Araby Company is beneficial as: it reduces paper work drawbacks, makes search and data collection easier, and reduces human errors in data entry. A direct interacting system between machine operator and engineers has been established. This system is a real-time and accurate data interchange, which acts on the product defects, stoppages and breakdowns. The collected data from machine and operator has been introduced to the MES through wireless communication. The I-MES increase the productivity as they reduce time waste at the beginning of the day that taken in tasks distribution. It at least in ideal day they safe the first shift, therefore this time is added to productivity, and machines availability. Also, they increase the productivity at the third shift via automatic supervision and control process on the machine.

I-MES reduce the machine stoppage time because of faster reporting and helpful in maintenance instructions where some times the problem is very simple. Power consumption and raw material losses have been reduced due to increase of products quality via injection process monitoring and helpful in quality inspection. I-MES reduce machine operator errors in counting and packaging process.

I-MES have some disadvantages

Qualitative issues difficult to quantify: Several of the benefits that are observed with I-MES are of a qualitative nature that is difficult to quantify. *Uncertainty:* Introducing a new system such as I-MES is related with a high level of uncertainty. Estimations have to be done in both costs and benefits. *Risk exposure:* There is also a risk in the stability of the I-MES. A breakdown in the system can have disastrous consequences. The conducted research could be expanded in the future to: Apply new artificial intelligence techniques in classification and control of machines. Cloud computing system may be used in data storage. Full scale I-MES implementation will be used over all El-Araby factories group. Apply six sigma techniques on quality operation management. Study more defects and breakdowns which affect the production operation performance. Apply machine vision techniques in quality inspection.

References

1. MESA: White Paper #01: The Benefits of MES: A Report from the Field. MESA International (1997). http://www.mesa.org/knowledgebase/details.php?id¼48. Accessed 12 Dec 2016
2. ANSI/ISA-95.00.03-2013: Enterprise-Control System Integration, Part 3: Activity Models of Manufacturing Operations Management. ISA–The Instrumentation, Systems, and Automation Society, Research Triangle Park (2005)
3. Eren, H.: 8 Standards in process control and automation. In: Instrument Engineers' Handbook, Volume 3: Process Software and Digital Networks 3, p. 155 (2016)
4. Choi, B.K., Kim, B.H.: MES (manufacturing execution system) architecture for FMS compatible to ERP (enterprise planning system). Int. J. Comput. Integr. Manuf. **15**, 274–284 (2002)
5. Elliott, R.F.: Manufacturing Execution System (MES): An Examination of Implementation Strategy. Master the Faculty of California Polytechnic State University, San Luis Obispo (2013)
6. Yin, S., Xie, X., Lam, J., Cheung, K.C., Gao, H.: An improved incremental learning approach for KPI prognosis of dynamic fuel cell system. IEEE Trans. Cybern. **46**(12), 3135–3144 (2016)
7. Jeon, B.W., et al.: An architecture design for smart manufacturing execution system. Comput. Aided Des. Appl. **14**(4), 472–485 (2017)
8. Kokina, J., Davenport, T.H.: The emergence of artificial intelligence: how automation is changing auditing. J. Emerg. Technol. Account. **14**(1), 115–122 (2017)
9. Li, B.H., Hou, B.C., Yu, W.T., Lu, X.B., Yang, C.W.: Applications of artificial intelligence in intelligent manufacturing: a review. Front. Inf. Technol. Electron. Eng. **18**(1), 86–96 (2017)
10. Witten, I.H., Frank, E., Hall, M.A., Pal, C.J.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, Burlington (2016)
11. Martynenko, A.: Artificial Intelligence: Is it a Good Fit for Drying?. Taylor & Francis, Routledge (2017)
12. Kumar, S., Nagpal, M.P.: Comparative Analyisis of P, Pi, Pid and Fuzzy Logic Controller for Tank Water Level Control System (2017)

13. Barkana, B.D., Saricicek, I., Yildirim, B.: Performance analysis of descriptive statistical features in retinal vessel segmentation via fuzzy logic, ANN, SVM, and classifier fusion. Knowl. Based Syst. **118**, 165–176 (2017)
14. Suwannasri, S., Sirovetnukul, R.: The defects reduction in injection molding by fuzzy logic based machine selection system. Int. J. Mech. Aerosp. Ind. Mechatron. Manuf. Eng. **7**(2), 255–263 (2013)
15. Salimi, A., Subas, M., Buldu, L., Karatas, C.: Prediction of flow length in injection molding for engineering plastics by fuzzy logic under different processing conditions. Iran. Polym. J. **22**, 33–41 (2013)
16. Harhalakis, G., et al.: Implementation of rule-based information systems for integrated manufacturing. IEEE Trans. Knowl. Data Eng. **6**(6), 892–908 (1994)
17. Sahm III, V.A., Hansen, C.: Plastic injection molding process. U.S. Patent No. 7,959,844 (2011)
18. Sadek, A.: Design and implementation of an expert system for monitoring and management of a web based industrial applications. Master, Ain Shams, Egypt (2013)
19. Oviedo, J.J.E., Vandewalle, J.P., Wertz, V.: Fuzzy Logic, Identification and Predictive Control. Springer Science & Business Media, London (2006)
20. Zadeh, L.A.: Fuzzy logic and the calculus of fuzzy if-then rules. In: Proceedings of the 22nd International Symposium on Multiple-Valued Logic, 27–29 May 1992, Sendai, Japan, p. 480 (1992)
21. Mahmoud, M.I., Ammar, H.H., Hamdy, M.M., Eissa, M.H.: Production operation management using manufacturing execution systems (MES). In: 2015 11th International Computer Engineering Conference (ICENCO), pp. 111–116. IEEE, December 2015

Improving Land-Cover and Crop-Types Classification of Sentinel-2 Satellite Images

Noureldin Laban[1](✉), Bassam Abdellatif[1], Hala M. Ebeid[2],
Howida A. Shedeed[2], and Mohamed F. Tolba[2]

[1] Data Reception and Analysis Division,
National Authority for Remote Sensing and Space Science, Cairo, Egypt
{nourlaban,bassam.abdellatif}@narss.sci.eg

[2] Faculty of Computer and Information Sciences,
Ain Shams University, Cairo, Egypt
{halam,Hoveyda.Saber,fahmytolba}@cis.asu.edu.eg

Abstract. Land cover and crop-types classification are of great importance for monitoring agricultural production and land-use patterns. Many classification approaches have used different parameters settings. In this paper, we investigate the modern classifiers using the most effective parameters to improve the classification accuracy of the major crops and land covers that exist in Sentinel-2 images for Fayoum region of Egypt. Four major crop-types and four major land-cover types are classified. This paper investigates the k-Nearest Neighbor (k-NN), Artificial Neural Network (ANN), Support Vector Machine (SVM), and Random Forest (RF) supervised classifiers. The experimental results show that the SVM and the RF report more robust results. The k-NN reports the least accuracy especially for crop types. The RT, K-NN, ANN, and SVM record 92.7%, 92%, 92.1% and 94.4% respectively. The SVM classifier out-performs the k-NN, ANN and RF classifiers.

Keywords: Artificial intelligence · Crop-types classification · Egypt
Remote Sensing (RS) · Satellite images · Sentinel-2

1 Introduction

Land cover and crop-types classification have become the vital usages of satellite image classification. It refers to the procedure through which different crop-types and land cover are discriminated from imagery based on their spectral behavior throughout the radiometric spectrum [1]. Using remote sensing imagery for crop classification over large areas has been broadly investigated recently [1–4]. Landsat-8 and Sentinal-2 satellites are the most recent satellites used in crop classification [1,5–8]. Time series of satellite images have been used as a related approach for classification or monitoring of agricultural crops for counties that suffer from clouds and rains most of year [1,3,9].

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 449–458, 2018.
https://doi.org/10.1007/978-3-319-74690-6_44

There are many classification techniques that have been used in crop classification. The most popular and efficient approaches for land cover classification are; ensemble based and deep learning. The K-nearest neighbor (K-NN) is a simple algorithm that stores all available cases and relegates incipient cases predicated on a kindred attribute measure. It was used as a reference approach to compare with other approaches. It used widely in remote sensing image classification [9–11]. Support Vector Machines are appropriate for remote sensing classification applications, simply for the fact that they need small training sets, on which it can give a good generalization [2,4,12]. Random Forest has been used efficiently in satellite image classification in recent years [13–15]. Random forest has successfully been applied for crop classification and in addition they were shown to provide meaningful information on classification uncertainty that can be used to evaluate map quality [1,6,7,11,16,17]. Artificial Neural Network (ANN) have been demonstrated to provide excellent performance in the classification of remotely sensed images [10,18–21].

Pena et al. [1] improved the classification accuracy of fruit-tree crops using different classifiers by examine the effect of spectrotemporal indices derived from Satellite Image Time Series (SITS). Zhu et al. [3] improved crop-types classification using Support Vector Machine (SVM) by merging Landsat with MODIS Nadir Bidirectional Reflectance Distribution Function-Adjusted Reflectance data. Nasirahmadi et al. discussed k-Nearest Neighbor (k-NN) and Support Vector Machine (SVM) classifiers as a Bag-of-Feature model for the classification of 20 sweet and bitter almond varieties [4]. Low et al. [16] presented the Support Vector Machine (SVM) and the Random Forest (RF) to classify crop-types at the object-level using 71 RapidEye time series. Gilbertson et al. [6,7] use Decision Trees (DTs), k-Nearest Neighbour (k-NN), Support Vector Machine (SVM) and random forest (RF) supervised classifiers to compare between automated and manual feature selection for the differentiation of crops in a Mediterranean climate using Landsat-8 images. Shastry et al. [21] used support vector machine (SVM) for crop-types classification with datasets having continuous attributes.

The rest of this paper is organized in three sections. In Sect. 2, We introduce Fayoum area and its ground truth datasets and present the used classifiers. Section 3 shows and discusses the experimental results for different classifiers. Finally, conclusions are drawn in Sect. 4.

2 Materials and Classifiers

2.1 Study Area and Satellite Images

Fayoum is a depression or basin in the desert immediately to the west of the Nile south of Cairo. The extent of the basin area is estimated at between 1,270 km^2 and 1700 km^2. The basin floor comprises fields watered by a channel of the Nile, the Bahr Yussef, as it drains into a desert depression to the west of the Nile Valley. The total area of the region is about 2000 km^2 with a diversity of different land cover types and agricultural crops as shown in Fig. 1.

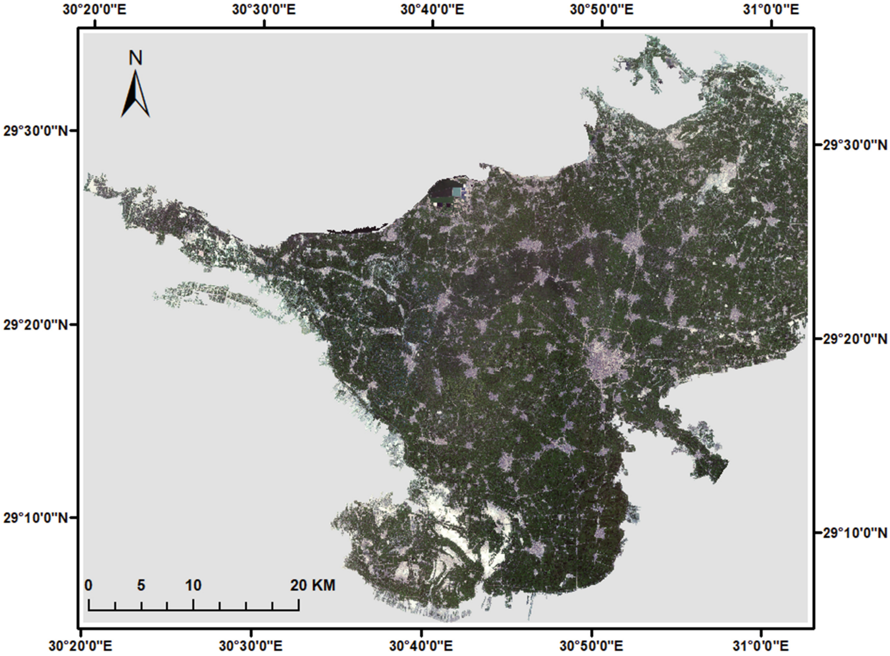


Fig. 1. Fayoum region

We addressed the problem of classify the crop-types and the land cover by using satellite images acquired by the Sentinel-2 satellite with a 10-m spatial resolution during March 2016. Satellite images formed from 10 bands for each pixel, radiometric and geometric correction have been applied to the satellite images. Also, all bands of the satellite images have been re-sampled to 10-m spatial resolutions.

2.2 Ground Truth Datasets

We have collected the ground truth datasets though February and March 2016. We have divided the ground truth data into 50% training dataset and 50% testing dataset as in Table 1. Each dataset is formed of a group of points shape files representing the target classes. We have four classes for crops namely: "sugar beet", "wheat", "trees" and "clover". Also, we have three Land Cover classes namely: "bare land", "water" and "urban" beside a class for background as shown in Figs. 2 and 3.

We get the intersection between the geospatial shape file for each class with the raster satellite image data. We get the labeled vector array for the two data sets; training data and testing data. Each sample value represents the 10-valued reflectance array of each pixel.

Table 1. Number of training and testing samples for each class.

#	Classes	Training samples	Testing samples
1	Wheat	122	122
2	Water	27	27
3	Urban	35	35
4	Trees	34	34
5	Sugar beet	13	14
6	Bare land	22	23
7	Clover	80	80
8	Background	21	19
Total samples		354	354

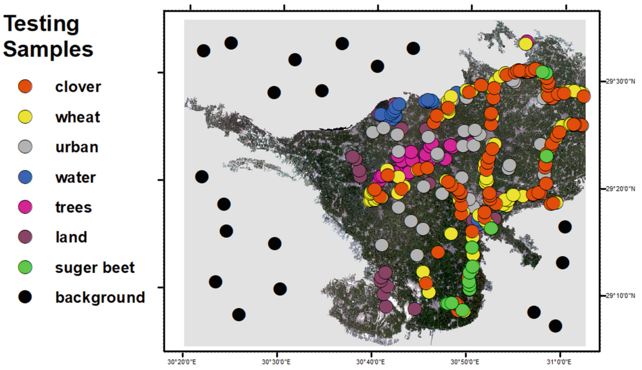


Fig. 2. Distribution of testing dataset.

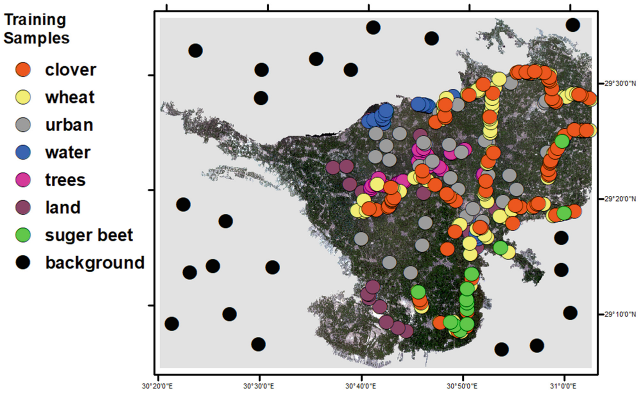


Fig. 3. Distribution of training dataset.

2.3 Classifiers

Crop-types classification was carried out for eight classes using four different classifiers and we investigated different classification parameters for each classifier in order to identify the efficient classification procedures. Since the different classification techniques differ in their ability to leverage nonlinear or otherwise complex relationships between features and crop-types, we used four different statistical and machine learning techniques that are representative the state-of-the-art image classification.

K-Nearest Neighbor (k-NN) is supervised learning. It finds the nearest k samples from the training data to the query sample. New test samples are classified according to the most similar class based on their distance. Euclidean distance is the most popular technique used to find the nearest neighbors [21]. Support Vector Machine (SVM) classifier is a non-parametric supervised classification derived from the statistical learning theory. Quadratic SVM training algorithm maps the training data into higher dimensional space and finds the optimal hyperplanes that separate the classes with minimum classification errors [12]. Random Forest (RF) classifier grows an ensemble of binary decision trees by selecting a fraction of bootstrap samples out of input data and choosing randomly a subset of explanatory variables for each split [14]. Artificial Neural Network (ANN) is made up of nodes arranged in layers namely input, hidden layers, and the output. Each node contain activation functions. Input layer presents input data pattern. Hidden nodes learn the input pattern through weighted connections. Output nodes check how the network is responding to the information it has learned [17].

3 Experimental Results

3.1 Experiments Setup

All the experiments are conducted on the same computer with 72 core Intel Xeon Phi Processor 7290 @ 2.50 GHz and 256 GB RAM. We also use scikit-learn python library as an open source, simple and efficient tool for data mining and analysis. We use also the Geospatial Data Abstraction Library (GDAL) as a computer software library for reading, processing and writing raster and vector geospatial data formats.

3.2 Results

k-Nearest Neighbor (k-NN) is a method used for classification. It is calculated the Euclidean distance d between the training examples x and y as follows in Eq. 1:

$$d = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{1}$$

The value for the k closest training examples is varied during the experiment to determined the impact of changing it on the classification accuracy.

Figure 4 shows the correct recognition rate as a function of k values. The best accuracy recorded at k = 3 as shown in Fig. 4.

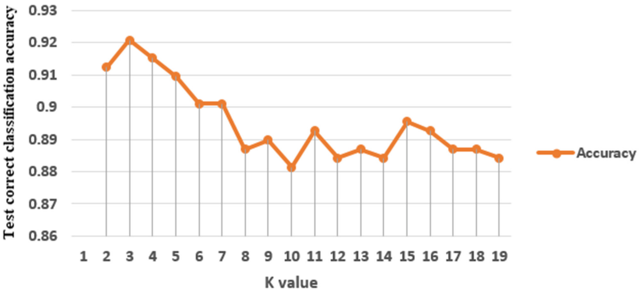


Fig. 4. Classification accuracy versus the parameter k value of K-NN.

Random Forest (RF). In a random forest, each tree is independent of the other trees in the forest, so that the training and testing procedures are in parallel [22]. The estimated probability for predicting class z for a sample is

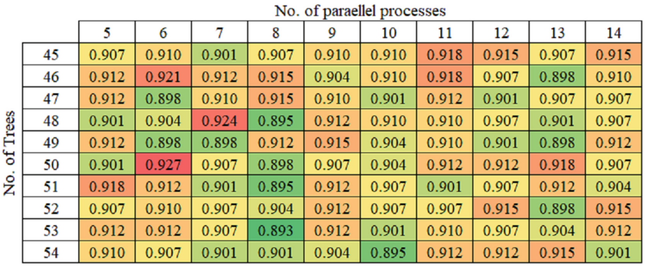
$$P(z|x) = \frac{1}{K}\sum_{k=1}^{K} P_k(z|x) \tag{2}$$

where $P(z|x)$ is the estimated density of the class labels of the k^{th} tree and K is the number of trees in the forest. The decision function of the forest is given by

$$C(x) = argmax_{j\in Z} P(z|x) \tag{3}$$

We examine two important parameters and their effects on accuracy. First one is the number of trees in the forest and second one is the number of parallel processes that investigate the forest. Table 2 shows the recognition accuracy for varying the number of parallel process against varying the number of trees in the forest. The test correct rate can increase significantly by increasing the number

Table 2. Classification accuracy for varying the number of parallel processes and the number of trees in the forest.



	No. of paraellel processes									
No. of Trees	5	6	7	8	9	10	11	12	13	14
45	0.907	0.910	0.901	0.907	0.910	0.910	0.918	0.915	0.907	0.915
46	0.912	0.921	0.912	0.915	0.904	0.910	0.918	0.907	0.898	0.910
47	0.912	0.898	0.910	0.915	0.910	0.901	0.912	0.901	0.907	0.907
48	0.901	0.904	0.924	0.895	0.912	0.910	0.910	0.907	0.901	0.907
49	0.912	0.898	0.898	0.912	0.915	0.904	0.910	0.901	0.898	0.912
50	0.901	0.927	0.907	0.898	0.907	0.904	0.912	0.912	0.918	0.907
51	0.918	0.912	0.901	0.895	0.912	0.907	0.901	0.907	0.912	0.904
52	0.907	0.910	0.907	0.904	0.912	0.907	0.907	0.915	0.898	0.915
53	0.912	0.912	0.907	0.893	0.912	0.901	0.910	0.907	0.904	0.912
54	0.910	0.907	0.901	0.901	0.904	0.895	0.912	0.912	0.915	0.901

of parallel processes and the number of tree until threshold value. The correct recognition rate archives 92.7% with number of trees equals 50 and number of parallel processes equals 6.

Artificial Neural Network (ANN). An ANN is a feedforward neural network which maps the input features into output through one or more hidden layers between the input and output layers. We investigate the most effective parameter in ANN which is the number of hidden layers and the number of neurons in each. Table 3 shows the recognition accuracy for varying the number of hidden layer and the number of neurons in each hidden layer. The correct recognition rate archives 92.1% with number of hidden layers equal 4 and number of neurons equals 144 in each hidden layer.

Table 3. Classification accuracy for varying the number of Hidden layers and the number of neurons in each hidden layer of ANN.

	No. of Hidden Layers				
No. of Neurons in each layer	1	2	3	4	5
140	0.116	0.746	0.893	0.853	0.777
142	0.240	0.469	0.912	0.458	0.777
144	0.065	0.523	0.582	0.921	0.893
146	0.314	0.703	0.514	0.588	0.489
148	0.376	0.209	0.407	0.740	0.229
150	0.113	0.672	0.763	0.839	0.853
152	0.226	0.732	0.489	0.605	0.912
154	0.511	0.729	0.805	0.791	0.658
156	0.520	0.483	0.884	0.822	0.506
158	0.342	0.489	0.446	0.915	0.444
160	0.579	0.427	0.802	0.907	0.452

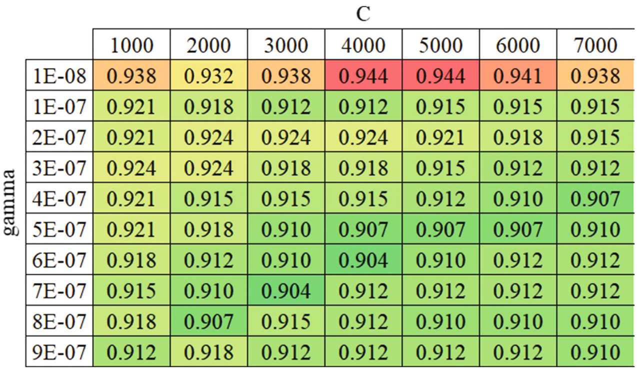
Support Vector Machine (SVM). The C and γ parameters play an important role for the nonlinear Support Vector Machine (SVM) with a kernel Gaussian radial basis function. The dual Lagrangian formulation of the soft 1-norm SVM reduces to the following quadratic program [23]:

$$\max_{\alpha_i} : \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j X_i^T X_j \tag{4}$$

$$subjected \quad to : \sum_{i=1}^{n} y_i \alpha_i = 0 \qquad 0 \leq \alpha_i \leq C, \quad i = 1....n \tag{5}$$

The experimental are carried out the determine the impact of changing the value of γ and C parameters. Table 4 shows the recognition accuracy for varying the value of gamma and C parameters. The correct recognition rate archives 94.4% with γ = 1E−08 and C = 4000.

Table 4. Classification accuracy for varying the value γ (gamma) and C parameters of SVM.



	C						
gamma	1000	2000	3000	4000	5000	6000	7000
1E-08	0.938	0.932	0.938	0.944	0.944	0.941	0.938
1E-07	0.921	0.918	0.912	0.912	0.915	0.915	0.915
2E-07	0.921	0.924	0.924	0.924	0.921	0.918	0.915
3E-07	0.924	0.924	0.918	0.918	0.915	0.912	0.912
4E-07	0.921	0.915	0.915	0.915	0.912	0.910	0.907
5E-07	0.921	0.918	0.910	0.907	0.907	0.907	0.910
6E-07	0.918	0.912	0.910	0.904	0.910	0.912	0.912
7E-07	0.915	0.910	0.904	0.912	0.912	0.912	0.912
8E-07	0.918	0.907	0.915	0.912	0.910	0.910	0.910
9E-07	0.912	0.918	0.912	0.912	0.912	0.912	0.910

3.3 Overall Classification Performance

The four classifiers used in this paper have a very near overall performance, providing classification accuracy that in the realm of remote sensing-based classifications, are commonly considered as good. The SVM has a slightly better performance than the others classifier methods using Radial Basis Kernel function. The SVM model reflected significantly the spatial differentiation of classes with small training data sets. RF has achieved nearby results from SVM but with more resources in terms of processing time and memory allocated. ANN has more oscillated results according to initial parameters of ANN. It also consume large amount of resources especially as network size increase. k-NN is more simple and direct classifier. It has an efficient use of resources but it decays quickly as k increase.

4 Conclusion

We addressed the classification accuracy of four crops and four land cover by investigating best parameters values for the state-of-the-art classifier namely: k-NN, RF, ANN and SVM. We use the complete spectral resolution of a Sentinel-2 Satellite Images corresponding to the 2016 winter season of the crops of interest and the intended land covers for Fayoum region of Egypt. For all the classifiers, the overall results were good (both recall and precision is greater than 90%). Both SVM and RF show robust accuracy as the accuracy remains high with slight change of parameters value where k-NN and ANN show wide changes in accuracy with slight changes in parameters value. Land cover types report more accuracy than Crops types as it is more specific spectral signature. SVM shows more discriminating power for crops while k-NN shows the least one. Although it achieve good results with simple land cover as water, background and urban. By comparing classification results of different combinations among the four classifiers approaches with broad scan of effective parameters on classification accuracy, we notice to what extent these parameters affect crops and land cover classification. We found that the best classification results 94.4% was achieved by SVM with parameters $C = 4000$ and $\gamma = 1E - 08$. In the future, we would like to get more improvement of classification using a hybrid approach using strength points in each classifier.

Acknowledgments. This work was supported in part by the GEF/World Bank Project “Regional Co-ordination for Improved Water Resources Management and Capacity Building” alongside The National Authority for Remote, Sensing and Space Science, Egypt.

References

1. Pena, M.A., Liao, R., Brenning, A.: Using spectrotemporal indices to improve the fruit-tree crop classification accuracy. ISPRS J. Photogram. Remote Sens. **128**, 158–169 (2017)
2. Waldhoff, G., Lussem, U., Bareth, G.: Multi-data approach for remote sensing-based regional crop rotation mapping: a case study for the Rur catchment, Germany. Int. J. Appl. Earth Obs. Geoinf. **61**, 55–69 (2017)
3. Zhu, L., Radeloff, V.C., Ives, A.R.: Improving the mapping of crop types in the Midwestern U.S. by fusing Landsat and MODIS satellite data. Int. J. Appl. Earth Obs. Geoinf. **58**, 1–11 (2017)
4. Nasirahmadi, A., Miraei Ashtiani, S.H.: Bag-of-feature model for sweet and bitter almond classification. Biosyst. Eng. **156**, 51–60 (2017)
5. Pena, M.A., Brenning, A.: Assessing fruit-tree crop classification from Landsat-8 time series for the Maipo Valley, Chile. Remote Sens. Environ. **171**, 234–244 (2015)
6. Gilbertson, J.K., van Niekerk, A.: Value of dimensionality reduction for crop differentiation with multi-temporal imagery and machine learning. Comput. Electron. Agric. **142**, 50–58 (2017)
7. Gilbertson, J.K., Kemp, J., van Niekerk, A.: Effect of pan-sharpening multi-temporal Landsat 8 imagery for crop type differentiation using different classification techniques. Comput. Electron. Agric. **134**, 151–159 (2017)
8. Sirsat, M.S., Cernadas, E., Fernández-Delgado, M., Khan, R.: Classification of agricultural soil parameters in India. Comput. Electron. Agric. **135**, 269–279 (2017)
9. Coniu, T., Groza, A.: Improving remote sensing crop classification by argumentation-based conflict resolution in ensemble learning. Expert Syst. Appl. **64**, 269–286 (2016)
10. Pathan, S., Prabhu, K.G., Siddalingaswamy, P.C.: Techniques and algorithms for computer aided diagnosis of pigmented skin lesions - a review. Biomed. Sign. Process. Control **39**, 237–262 (2018)
11. Piiroinen, R., Heiskanen, J., Mõttus, M., Pellikka, P.: Classification of crops across heterogeneous agricultural landscape in Kenya using AisaEAGLE imaging spectroscopy data. Int. J. Appl. Earth Obs. Geoinf. **39**, 1–8 (2015)
12. Zheng, B., Myint, S.W., Thenkabail, P.S., Aggarwal, R.M.: A support vector machine to identify irrigated crop types using time-series Landsat NDVI data. Int. J. Appl. Earth Obs. Geoinf. **34**(1), 103–112 (2015)
13. Wu, Z., Lin, W., Zhang, Z., Wen, A., Lin, L.: An ensemble random forest algorithm for insurance big data analysis. In: 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), vol. 5, pp. 531–536 (2017)

14. Li, L., Solana, C., Canters, F., Kervyn, M.: Testing random forest classification for identifying lava flows and mapping age groups on a single Landsat 8 image. J. Volcanol. Geoth. Res. **345**, 109–124 (2017)
15. Medeiros, S.C., Hagen, S.C., Weishampel, J.F.: A random forest model based on lidar and field measurements for parameterizing surface roughness in coastal modeling. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. **8**(4), 1582–1590 (2015)
16. Low, F., Michel, U., Dech, S., Conrad, C.: Impact of feature selection on the accuracy and spatial uncertainty of per-field crop classification using support vector machines. ISPRS J. Photogram. Remote Sens. **85**, 102–119 (2013)
17. Chen, W., Pourghasemi, H.R., Kornejady, A., Zhang, N.: Landslide spatial modeling: introducing new ensembles of ANN, MaxEnt, and SVM machine learning techniques. Geoderma **305**, 314–327 (2017)
18. Taravat, A., Del Frate, F., Cornaro, C., Vergari, S.: Neural networks and support vector machine algorithms for automatic cloud classification of whole-sky ground-based images. IEEE Geosci. Remote Sens. Lett. **12**(3), 666–670 (2015)
19. Barreto, T.L., Rosa, R.A., Wimmer, C., Moreira, J.R., Bins, L.S., Cappabianco, F.A.M., Almeida, J.: Classification of detected changes from multitemporal high-res Xband SAR images: intensity and texture descriptors from superpixels. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. **9**(12), 5436–5448 (2016)
20. Mountrakis, G., Im, J., Ogole, C.: Support vector machines in remote sensing: a review. ISPRS J. Photogram. Remote Sens. **66**(3), 247–259 (2011)
21. Shastry, K.A., Sanjay, H.A., Deexith, G.: Quadratic-radial-basis-function-kernel for classifying multi-class agricultural datasets with continuous attributes. Appl. Soft Comput. J. **58**, 65–74 (2017)
22. Dong, Y., Du, B., Zhang, L.: Target detection based on random forest metric learning. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. **8**(4), 1830–1838 (2015)
23. Paul, S., Magdon-Ismail, M., Drineas, P.: Feature selection for linear SVM with provable guarantees. Pattern Recogn. **60**, 205–214 (2016)

GPU-Based CAPSO with N-Dimension Particles

Shafaatunnur Hasan[1,2](✉), Amantay Bilash[1,2], Siti Mariyam Shamsuddin[1,2], and Aboul Ella Hassanien[3]

[1] UTM Big Data Centre, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia
{shafaatunnur,mariyam}@utm.my
[2] Faculty of Computing, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia
[3] Faculty of Computers and Information, Cairo University, Cairo, Egypt
aboitcairo@gmail.com

Abstract. Today we are living in a world that is surrounded with information obesity which is also known as Big Data. Big data deals with zeta bytes of data flown from variety sources, and cannot be processed or analyzed using traditional procedure. Due to this, there is an increasing interest of researchers in using low cost GPUs for various applications that require intensive parallel computing to solve complex problems much faster. Various machine learning algorithms have been developed to obtain the optimal solutions with various data complexity. However, for big data problems, new machine learning algorithms need to be developed to deal with zeta bytes data problems. Centripetal accelerated particle swarm optimization (CAPSO) is the recent machine learning algorithm to enhance the convergence speed, accuracy and global optimality for optimization problems. However, the convergence speed of CAPSO is limited for small number of particles only. Hence, this research proposes improved CAPSO by implementing this algorithm on GPU platform through CUDA programming to handle N-dimensional scale of particles. Since CAPSO is intrinsically parallel processing, thus it can be effectively implemented on Graphics Processing Units (GPUs) according. The proposed GPU-based CAPSO was tested on various multi modal test functions and the results have proven that the proposed GPU-based CAPSO has successfully reduced the execution time with various particles dimensions compared to CPU-based CAPSO.

Keywords: Big data · N-dimensional particles · PSO · GPU computing
Machine learning and optimization

1 Introduction

Today we live in a world that called Big Data era which has about a billion transistors per person, a world with around 4 billion mobile phone subscribers and about 30 billion radio frequency identification tags produced all over the world in two years. These sensors all produce data across the whole ecosystems, from social networks, traffic flow sensors, broadcast audio streams, banking transactions, scans of government documents, GPS trails financial market data, web server logs, satellite imagery; with common activity which is data generation. However, due to its voluminous and non-uniform

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 459–467, 2018.
https://doi.org/10.1007/978-3-319-74690-6_45

nature that leads to big data, advance computing platform, infrastructure and algorithms are needed for deep analytics processing and optimization for better insight and foresight that can't be processed or analyzed using traditional processes or instruments.

Thus many advance computing platforms have been improved for advance analytics such as Graphics processing unit (GPU). GPU is one of the technology advancements of also referred to visual process units (VPU) are computer boards which were originally fabricated as a graphics processing device. It has now broadened its application to generate purpose high performance computing. The GPU, over the past few years has had increasing improvements in its performance and capabilities. This can be attributed to the fact that the GPU is a highly parallel programmable processor that facilitated complex programmability in diverse fields and areas. The distinguishing feature of the GPU which is the parallelism is gradually becoming the future of computing as future microprocessor development will be directed towards the parallel architecture by adding cores rather than increasing single thread performance.

The future of computing is parallelism in tandem to the advancement of the hardware and software for advance intelligent solutions and wealth creation of digital skills. Hence, deep analytics processing and optimization such as machine learning and optimization algorithms must be well blended dealing with big data processing and complexity in which many organizations nowadays are facing more and more big data challenges. They have access to an abundant of information, but they don't know how to get value from it because it is stored in its most raw form or in a semi-structured or unstructured format; and as a result, they don't even know whether it's worth keeping. Thus practical applications of standard optimization and analytics have become the main concern around industry and scientific research whenever dealing with big data.

Machine Learning (ML) is a branch of Artificial Intelligence (AI) concerned with many learning algorithms and problems. Different ML algorithms have been successfully employed to solve real-life problems. The goal of ML research is computer learning based on training data to recognize complex patterns of datasets, or to make intelligent decisions based on data. In ML, optimization provides a valuable framework for thinking about, formulating and solving many problems. Optimization problems have located at the heart of most ML approaches. Many algorithms from the class of exact and approximate optimization algorithms have been presented to deal with ML applications. However, exact optimization algorithms such as dynamic programming, branch-and-bound and backtracking have shown good performance in addressing ML applications, they are not efficient in a high-dimensional search space. In the applications, the search space increases exponentially with the problem size, hence solving these problems using the algorithms (such as exhaustive search) is not practical. Therefore, many researchers are interested in utilizing approximate algorithms like meta-heuristic algorithms in this regard.

2 Population-Based Meta Heuristics Algorithms (MHA)

Meta-heuristic algorithms are widely used to solve intractable problems with incomplete or imperfect information. It was designed to generate a lower-level procedure or heuristic that may provide a sufficiently better solution. The main difference between meta-heuristic and optimization algorithms is that meta-heuristics are not assuring that the global optimal will be achieved. In meta-heuristics the finding of solution is depending on the set of random generated variables. With relatively less computational effort it can find good solutions over a large set of feasible solutions. Meta-heuristic approaches are very suitable for addressing non-deterministic polynomial-hard (NP-hard) optimization. Therefore, many scientists in this field are paying more attention on solving disadvantages of optimization algorithms by meta-heuristics methods.

MHA can be classified by different ways. For example, [1] categorized algorithm on nature-inspired, population-based and dynamic functions. The examples are Particle Swarm Optimization [2], Genetic Algorithm [3], Artificial Immune System [4], Ant Colony Optimization [5], Artificial Bee Colony [6], Imperialistic Competitive Algorithm [7], Gravitational Search Algorithm [8] and Charged system search [9].

2.1 Particle Swarm Optimization (PSO)

PSO is a heuristic computational method, which is based on flock intelligence. It was proposed in 1995 by social psychologist James Kennedy and professor and chairman of electrical and computer engineering Russell C. Eberhart. The basic idea was originally inspired by simulation of the social behavior of animals such as bird flocking, fish schooling and so on. If any particle of the group can determine a desirable way to fly, then the other particles of the group will follow it rapidly. In PSO, each swarm is "flying" through the solution space to find the best solution. Only the fixed number of agents are involved in the search area. Throughout the each iteration every agent is responsible for updating its own velocity and position based on the personal best location and the best location of whole swarm. The position of each particle stands for a candidate solution and the particles are updating their current states based on the following equations:

$$v_{id}^{(k+1)} = v_{id}^{k} + c_1 r_1 \left(pbest_{id}^{k} - x_{id}^{k}\right) + c_2 r_2 \left(gbest_{id}^{k} - x_{id}^{k}\right) \tag{1}$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \tag{2}$$

where

v_{id}^{k} - the velocity of agent id at step k,

x_{id}^{k} - the position of agent id at step k,

$pbest_{id}^{k}$ - the personal best of agent id at step k,

gbest_{id}^{k} - the global best of agent so far,

c_1 - the cognitive acceleration coefficient so named for its term's use of the personal best,

c_2 - the social acceleration coefficient,

r_1 and r_2 are vectors of pseudo-random numbers with components selected from uniform distribution U(0, 1) at iteration k.

The position and velocity of each particle are initialized randomly in the first iteration. Every agent is evaluated by fitness function and the fitness value is compared with preceding personal best value of the particle and the best value in whole swarm. If the current fitness value smaller than preceding personal best value, we should set the current fitness value as a personal best. If the current fitness value smaller then global best value, we update the global best. Thus, the updating process is influenced by personal and social impact.

The common disadvantages of current exist evaluations are they either perform well on real optimization problems or binary optimization problems. To address this problem Beheshti Z. and Shamsuddin S.M. introduced new scheme of PSO merged with Newton's rules of motion which so-called centripetal accelerated particle swarm optimization (CAPSO) [10].

2.2 Centripetal Particle Swarm Optimization (CAPSO)

CAPSO increased the convergence speed and exploration ability. Newtonian laws represent the movement of objects. If an object moves from position x_1 to x_2 during the time step △t, and the velocity changing from initial velocity v_1 to v_2 during the time, the moving object will be accelerated. The acceleration is equal to:

$$a = \frac{v_2 - v_1}{\triangle t} \tag{3}$$

Based on Eq. (3), the next velocity is obtained by Eq. (4) as:

$$v_2 = v_1 + a \ * \triangle t \tag{4}$$

Then, the length travelled by the object is as follows:

$$x_2 = x_1 + 1/2 \ * \triangle t^2 + v_1 \ * \triangle t \tag{5}$$

CAPSO prevails over the actual disadvantages associated with PSO through speeding up the convergence speed as well as preventing local optima. The PSO algorithm lack of ability of escapes from local optimum. In PSO, when a particle in local optimum, the personal best and current position are in the same local optimum, the second and last phrases of Eq. (1) tend to zero. This causes the next velocity have a tendency to zero and position does not change, as a result particle stay in the local optimum. CAPSO does apply typically the rules involving motions throughout movement to PSO to conquer the aforementioned problems. Particles update own velocity according to current velocity, acceleration and centripetal acceleration, as follow:

$$v_{id}(t+1) = v_{id}(t) + a_{id}(t) + A_{id}(t) \tag{6}$$

Where a_id(t) is the acceleration, as follow:

$$a\_id(t) = rand * (p\_id(t) - x\_id(t)) + rand * (p\_gd(t) - x\_id(t)) \tag{7}$$

Centripetal acceleration equal to:

$$A_{id}(t) = E_i(t) * \text{ rand } * \left(P_{id}(t) - P_{med,d}(t) - x_{id}(t)\right) \tag{8}$$

Where $P_{med,d}(t)$ is the current median position of particles in dimension d and $E_i(t)$ is the acceleration coefficient given by Eq. (10) as follows:

$$e_i(t) = fit_i(t) - GWfit(t) \tag{9}$$

$$E_i(t) = \frac{e_i(t)}{\sum_{j=1}^{N} e_j(t)} \tag{10}$$

$fit_i(t)$ is the fitness value of the particle i
$GWfit(t)$ is the worst fitness value discovered so far by the swarm

The next position will be:

$$x_{id}(t+1) = x_{id}(t) + \frac{1}{2} * a_{id}(t) + v_{id}(t+1) \tag{11}$$

So the difference between PSO and CAPSO is that in CAPSO algorithm quoted centripetal acceleration which helps it to escape from local optima.

3 CAPSO Implementation on CUDA Environment

The basic idea of implementation CAPSO on CUDA is to execute parallelizable part of CAPSO on CUDA and to keep sequential part of instructions executed on CPU. The code with single functional unit executed on the devices called kernel. Because transfer data between host and device is bottleneck in GPU computing a simple mechanism designed to reduce data transferring between grid and global memory is presented below (Fig. 1):

```
 Step 1:  Initialize parameters of particle on CPU
 Step 2:  Move parameters of particle on CPU
 Step 3:  Initialize particles according to parameters in GPU
 Step 4:  Evaluate fitness value
 Step 5:  Find best and worst fitness value
 Step 6:  If the fitness value is better than the best fitness value (pbest) achieved
          so far, set current value as the new personal best (pbest)
 Step 7:  Find the global best and global worst memory
 Step 8:  Find median position
 Step 9:  Calculate acceleration for each particles
Step 10:  For each particle, update velocity
Step 11:  For each particle, update position
Step 12:  Repeat step 4 to step 12 until maximum iteration
Step 13:  Move the best value found in global memory to CPU
```

Fig. 1. CAPSO-GPU Pseudocode

In this study, a GPU based CAPSO algorithm named CAPSO-GPU is proposed and employed in order to reduce execution time of CAPSO algorithm. The experiments are performed on the machine Intel® Core™ i5-3210 M CPU 2.5 Ghz with 6 GB RAM. The GPU device is NVIDIA GeForce 610 M with 2 GB RAM and 48 cores, GPU clock is 738 MHz with compute capability 2.1. The proposed CAPSO-GPU was tested by benchmark function as shown in Table 1.

Table 1. Benchmark function.

Test function	$[\text{Range}]^n$	F_{opt}
$F_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	$[-10, 10]^n$	0
$F_3(x) = \sum_{i=1}^{n} (\sum_{j=1}^{i} x_j)^2$	$[-100, 100]^n$	0

4 Experimental Results and Discussion on CAPSO-GPU

In order to see the relation between speed up and parameters of CAPSO and CAPSO-GPU in each test the number of particles was set up to 32, 64, 128, 256 and 512 for both CAPSO-GPU and CAPSO-CPU. The number of blocks and the number of threads per block are crucial parameters in GPU computing. To observe the impact of parameter settings to speed up of CAPSO-GPU, in each test the number of threads is set to different number and this number equal to exponential of 2 and less than number of particles. For example, if the number of particles are 32 on CAPSO-CPU then the number of particles in CAPSO-GPU are also equal to 32, then CAPSO-GPU was tested by 3 times: 4 threads per block, 8 threads per block and 16 threads per block. Therefore, the results are shown in Figs. 2 and 3 accordingly.

Figure 2 illustrates a significant difference on execution time between CAPSO-GPU and CAPSO-CPU as the number of particles is larger. For instance, CAPSO-GPU with 32 particles faster than CAPSO-CPU around 6 times, and with 512 particles CAPSO-GPU faster than CAPSO-CPU around 18 times.

Table 2. Convergence performance of CAPSO on F_2 and F_3

Convergence performance	F_2	F_3
Average best solution	3.39E-20	1.04E-15
Standard deviation	3.39E-20	1.60E-15
Median solution	2.27E-20	4.54E-16
Best solution	4.02E-21	2.68E-18
Execution time (ms)	1338.75	3530
Optimal solution (iteration)	100	100

As in Fig. 3, CAPSO-GPU speedup is 82 times faster than CAPSO-CPU; with 128 threads per block and number of particles equal to 512. This is due to the complexity of benchmark function, F_2 is higher than F_3. Hence, CAPSO-CPU and CAPSO-GPU

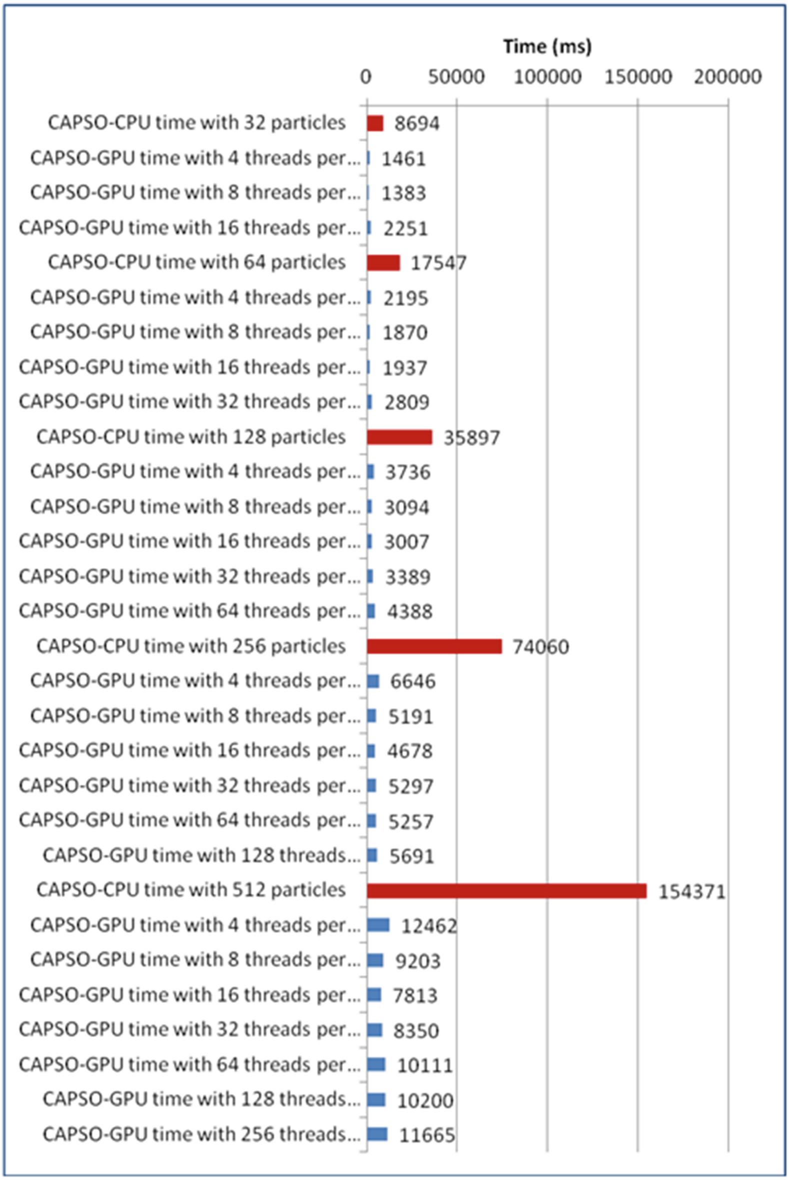


Fig. 2. The comparison of execution time of CAPSO-CPU and CAPSO-GPU on benchmark function, F_2

speedup is highly related to hardware specification, parameter setting and benchmark function complexity. For convergence performance, both functions, F_2 and F_3 are tested with thousand iterations. The result is shown in Table 2, respectively.

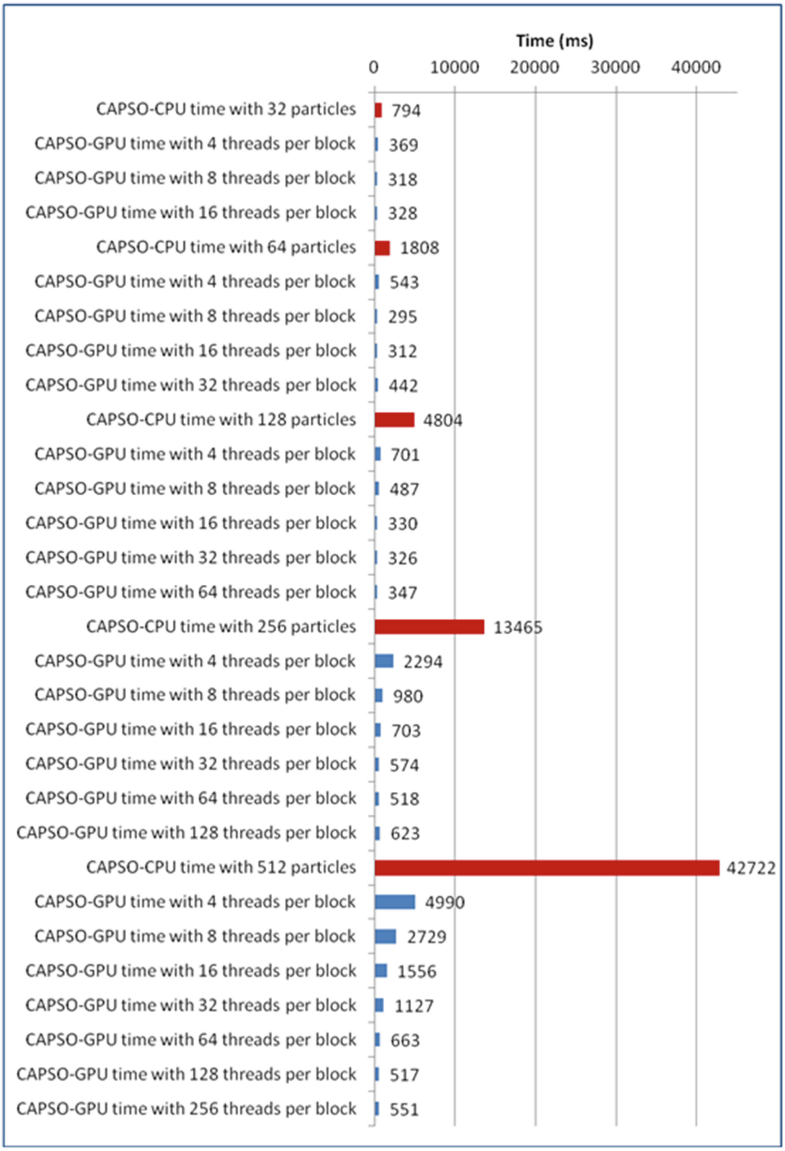


Fig. 3. The comparison of execution time of CAPSO-CPU and CAPSO-GPU on benchmark function, F_3

5 Conclusion

In this study, the performance of CAPSO-GPU algorithm is faster than CAPSO-CPU algorithm. The speed-up of CAPSO-GPU increases as the number of particles increases. The number of threads per block is very important parameter in GPU computing and it affects the performance of GPU. Generally, CAPSO-GPU get better speedup performance when the number of threads per block equal to half of number of particles. Furthermore, the convergence performance is feasible for F_2 and F_3. Both functions converge at near optimal solution of zero ($F_2 = 0$ and $F_3 = 0$) with 100 iterations, respectively.

Acknowledgements. The authors would like to thank the Universiti Teknologi Malaysia (UTM) for their support in Research and Development, UTM Big Data Centre and the Soft Computing Research Group (SCRG) for the inspiration in making this study a success. This work is supported by Ministry of Higher Education (MOHE) under Fundamental Research Grant Scheme (FRGS), Grant No. 4F802 and 4F786; and UTM under Research University Grant (RUG), Grant No. 17H62.

References

1. Dreo, J.: Dreaming of metaheuristics (2007). http://metah.nojhan.net
2. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of the 1995 Proceedings of IEEE International Conference on Neural Networks, Perth, Australia, 1942–1948 (1995)
3. Holland, J.H.: Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. U Michigan Press, Ann Arbor (1975)
4. Farmer, J.D., Packard, N.H., Perelson, A.S.: The immune system, adaptation, and machine learning. Physica D **22**(1–3), 187–204 (1986)
5. Dorigo, M., Maniezzo, V., Colorni, A.: Ant system: optimization by a colony of cooperating agents. IEEE Trans. Syst. Man Cybern. Part B Cybern. **26**(1), 29–41 (1996)
6. Karaboga, D.: An idea based on honey bee swarm for numerical optimization. Technical report-tr06, Erciyes University, Engineering Faculty, Computer Engineering Department (2005)
7. Atashpaz-Gargari, E., Lucas, C.: Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. In: IEEE Congress on Proceedings of the 2007 Evolutionary Computation, CEC 2007, 25–28 September 2007, pp. 4661–4667 (2007)
8. Rashedi, E., Nezamabadi-pour, H., Saryazdi, S.: GSA: a gravitational search algorithm. Inf. Sci. **179**(13), 2232–2248 (2009)
9. Kaveh, A., Talatahari, S.: A novel heuristic optimization method: charged system search. Acta Mech. **213**(3–4), 267–289 (2010)
10. Beheshti, Z., Shamsuddin, S.M.H.: CAPSO: centripetal accelerated particle swarm optimization. Inf. Sci. **258**, 54–79 (2013)

Machine Learning and Big Data Processing: A Technological Perspective and Review

Roheet Bhatnagar(✉)

Department of Computer Science and Engineering, Manipal University Jaipur,
Jaipur, Rajasthan, India
roheet.bhatnagar@jaipur.manipal.edu

Abstract. This paper discusses the role of Machine Learning (ML) based algorithms and methods in Big Data Processing & Analytics (BDA). ML and BDA are both evolutionary fields of computing and the developments in these fields are complementing each other. The ever changing data landscape in modern digital world have resulted in newer ways of data processing frameworks in order to get meaningful insights which are unprecedented. This paper presents a detailed review on latest developments in ML algorithms for Big Data Processing. In later section key challenges associated with application of ML based approaches are also discussed. ML based Big Data Processing has gained popularity and new developments are on the rise for efficient data processing. This field is witnessing unparalleled emergence of new methods and approaches for efficient data processing in order to discover interestingness for decision making. Thus, more and more ML based data processing approaches are being used for Big Data Processing. With the splurge data from different newer sources, heterogeneous nature of data, uncertain & unstructured data, the so called Big Data with all its characteristics (5 Vs) there is an ever increasing need to use approaches which aid in modelling and processing of these data, provide automated approach to data processing and so on. These type of new processing requirements have given a big boost to the development of new ML based methods for managing & processing them. The paper will be useful to the scholars who are researching in this interesting & challenging domain of ML and Big Data Processing.

Keywords: Machine Learning · Big Data Processing & Analytics
Decision making

1 Introduction

The world is witnessing a tremendous technological advancements and ever increasing need to understand data, because today 'Data is Power and Data is Money'. We are living in an era where we are witnessing unprecedented amount of data being generated from several unheard and unseen sources. We have technology developed to capture, manage and process these unforeseen data, but still

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 468–478, 2018.
https://doi.org/10.1007/978-3-319-74690-6_46

there are many challenges and issues which need to be tackled. Many researches are going on in these direction to better understand & have many meaningful insights from Big Data. Today in every field of study, be it Basic Sciences, Applied Sciences, Engineering, Social Sciences, Bio-Medical Sciences and so on we are dealing with Big Data. All of these fields are dealing with Large Scale datasets [1] and lot of work is being carried out to better harness and process Big Data, using domains like Machine Learning (ML) which holds tremendous potential in handling modern data challenges. According to one study [2], in 2011, digital information has grown nine times in volume in just 5 years and its amount in the world will reach 35 trillion gigabytes by 2020 [3].

The paper aims at discussing numerous issues related to humongous amount of data, their processing and analytics, current research focus and the future trends. It also talks about utilizing the machine learning approaches for Big Data Processing and highlights the current scenario in the domain from different perspectives. The paper makes many contribution and it is organized as follows. Section 2 concerned with advent of Big Data and recent developments in Big Data Processing & Analytics domain. Section 3 starts with a description of the evolution of Machine Learning and deals with assessment of the traditional machine learning techniques, their applications followed by many advanced learning methods as a direct outcome of recent researches carried out to efficiently perform Big Data Processing. Several issues, challenges and opportunities associated with Big Data Processing and the solutions proposed by different researchers are discussed in the Sect. 4. Section 5 concerned with future issues and challenges which need to be worked upon to get many new meaningful insights.

2 Big Data - An Introduction

"Big Data" refers to a collection of tools, techniques and technologies for working with data productively, at any scale. Increase in the storage capacity & advanced storage technologies, increased processing capacity of modern day computers and availability of large scale data – all have led to the development in Big Data Processing field. Modern times hardware and software technologies can manage, manipulate, process and analyze humongous amount of data as never before.

The explosion of the Internet, social media technology, devices and apps is creating a tsunami of data. Extremely large sets of data can be collected and analyzed to reveal patterns, trends and associations related to human behavior and interactions. Big data is being used to better understand consumer habits, target marketing campaigns, improve operational efficiency, lower costs, and reduce risk. International Data Corporation (IDC), a global provider of market intelligence and information technology advisory services, estimates that the global big data and analytics market will surge in times to come [4]. The challenge for businesses is how to make the best use of this wealth of information.

Some experts break down big data into three subcategories: (i) Smart data; (ii) Identity data; and (iii) People data [5]. Big data sets are so large that traditional processing methods often are inadequate [6]. Gartner's 2014 Hype Cycle, includes Big Data as technology of the future [7–9].

Today's Big Data may not be Big Tomorrow, since data is being continuously generated and we are flooded with data which need to be harnessed & processed to gain new insights. As such, traditional data processing tools which do not scale to big data will eventually become obsolete.

Everyone is processing Big Data, and trying to harness the benefits out of processing using various Big Data processing frameworks. Apache Hadoop and Spark are some of the popular frameworks and are very well-known, while there are others which are more niche in their usage, but have still managed to carve out respectable market shares and reputations [9]. Generally speaking, these frameworks can be categorized as Proprietary Frameworks and, Open Source Frameworks and both types are popular in industry. Hadoop, Spark, Flink, Storm, and Samza are some of the popular Open Source Big Data processing frameworks.

3 Machine Learning - A Brief Introduction

This section discusses the concepts of machine learning, its evolution, different ML techniques applications and finally discusses the Advanced Machine Learning techniques proposed in recent past. ML based problem solution is very much required in the field of Big Data Processing.

3.1 Machine Learning Techniques - Classification and Use

The concept of Machine Learning is not new in the field of computing, however due to ever changing nature of requirements of today's world it has come up in a new 'Avatar' all together. Now we find everyone talking of ML based solution strategies for a given problem set. ML is a subset of Artificial Intelligence, where computer algorithms are used to autonomously learn from data and information. With the rise of the internet, there is a lot of digital information being created - which means there is more data available for machines to analyse and 'learn' from [10]. Hence, as a result we see the resurgence of Machine Learning. Today, machine learning algorithms enable computers to communicate with humans, autonomously drive cars, write and publish sport match reports, and find terrorist suspects. Machine learning (ML) is the most growing field in computer science [11].

Classification [12,13], regression [14], topic modelling [15,16], time series analysis [16], cluster analysis [12,16,17], association rules [14,16], collaborative filtering [13,18,19], and dimensionality reduction [20,21] are some of the popular Machine learning techniques/methods. These are used to perform analytics and predict the future trends based on the existing patterns and correlations among data in the given dataset.

Agneeshwaran [22], proposed a maturity model for describing advanced analytics and it also distinguishes analytical tools into three generations of machine learning as follows [23]:

- 1st Generation Machine Learning (1GML) requires the data workload to fit into memory of a single machine. Such tools are restricted to vertical scaling which is a drawback when considering Big Data. Tools in this group were usually developed before Hadoop and are referred to as traditional analytical tools. (R, RapidMiner, KNIME, SAS, WEKA are some of the examples of 1GML tools).
- 2nd Generation Machine Learning (2GML) enhances 1GML with capabilities for distributed processing across Hadoop clusters. In contrast to 1GML, data remains at its location while the code execution is divided and processed on each required data node in parallel (Mahout (MapReduce) is an example).
- 3rd Generation Machine Learning (3GML) enhances 2GML with capabilities to efficiently perform distributed processing of iterative algorithms. This class is referred to as beyond Hadoop (Mahout (Spark/H2O/Flink), MLlib, H2O ML, Flink-ML SAMOA, MADlib are some of the examples).

ML has evolved tremendously from the classical Turing Test proposed by Alan Turing in 1950 to AlphaGo algorithm by Google DeepMind, Google Inc in 2016 [24].

Qiu et al. [25] in Table 1, provided the comparison of three subdomains of ML from different perspectives and outline the ML technologies for data processing.

Table 1. Comparison of machine learning technologies [25].

Learning types	Data processing tasks	Distinction norm	Learning algorithms	References
Supervised learning	Classification/ regression/ estimation	Computational classifiers	Support vector machine	[26]
		Statistical classifiers	Naive Bayes	[27]
			Hidden Markov model	[28]
			Bayesian networks	[29]
		Connectionist classifiers	Neural networks	[30]
Unsupervised learning	Clustering/ prediction	Parametric	K-means	[31]
			Gaussian mixture model	[32]
		Nonparametric	Dirichlet process mixture model	[33]
			X-means	[34]
Reinforcement learning	Decision making	Model-free	Q-learning	[33]
			R-learning	[33]
		Model-based	TD learning	[34]
			Sarsa learning	[35]

Supervised learning, unsupervised learning, and reinforcement learning are the three sub domains of Machine Learning [36]. A lot of development vis–vis the theory mechanisms and application services have been proposed for dealing with data tasks [37–39] in the above subdomains of ML. Spam Detection, Credit Card Fraud Detection, Digit Recognition, Speech Understanding, Face Detection, Shape Detection, Product Recommendation, Medical Diagnosis, Stock Trading, Customer Segmentation are some of the key applications of Machine Learning [33].

Big Data Processing Frameworks like Apache Spark have got Machine Learning libraries and components to apply ML on Big Data. Apache Spark is a general data processing framework and the various components of the framework are used by researchers for specific purpose across the globe [34]. MLlib is Spark's machine learning (ML) library. Its goal is to make practical machine learning scalable and easy. Similarly, Apache Flink is an open-source stream processing framework for distributed, high-performing, always-available, and accurate data streaming applications.

4 Big Data Processing

Big Data Processing is a focus area of research and many frameworks & techniques have been proposed in recent past by different researchers. Big Data has become important as many organizations both public and private have been collecting massive amounts of domain-specific information, which can contain useful information about problems such as national intelligence, cyber security, fraud detection, marketing, and medical informatics. Companies such as Google and Microsoft are analyzing large volumes of data for business analysis and decisions, impacting existing and future technology [35]. Big Data Analytics is helping organizations to improve business efficiency but then there are many challenges & issues associated with Big Data Processing & Analytics for each of the 5 Vs (Volume, Velocity, Variety Veracity, and Value) [25].

4.1 Issues, Challenges and Opportunities in Big Data Processing Using Machine Learning

Business needs are changing, and ever increasing & diversified data poses new challenges to the researchers. The Big Data Processing is gaining prominence as the world has realised the potential of discovering meaningful insights from unstructured data over the past few years now. So is true for Machine Learning algorithms which have been pushed to the forefront and are aiding in making more timely & accurate predictions. ML algorithms are used to realize the value of Big Data and to process the large data volume at high velocity than ever before witnessing tremendous & unprecedented changes. There is continuous development in the field of ML as well but still the Models Scalability and Distributed Computing [40] are some of the key challenges to ML implementations during Big Data Processing.

A typical ML system usually consists of a Data Pre-processing unit, Model builder & Evaluation unit and an Output unit. Data pre-processing is an important phase and poses new challenges, issues and opportunities to researchers & practitioners in the Big Data era. During pre-processing the raw data is transformed and it results in certain representation of data that can support effective ML applications [41].

Traditional data pre-processing and preparation relies on human interventions and is costlier and error prone; while with the application of ML algorithms on Big Data, opportunities have been created in reducing the reliance on human monitoring and supervision, as ML algorithms learn from massive, diverse and data in motion on their own.

Zhou et al. [40] and others have done an excellent review and the following section discusses each of the above pre-processing issues along with challenges & proposed solutions to mitigate the risk associated with them [40].

a. **Redundancy in Data:** It is also known as data duplication and it leads to inconsistent data which can be detrimental to ML based system. Techniques do exist for identifying duplicates in a given data set [42] but these traditional methods are not effective in case of Big Data. Techniques such as Dynamic Time Warping are much more efficient than traditional Euclidian Distance algorithms [43,44].
b. **Noisy Data:** Missing or incorrect values of data are one of the primary source of noise in data, which may severely hamper the outcome of applying analytics over the data set containing noise. Traditional mechanisms of removing noise from the data set fails in case of Big Data Processing due to their lack of scalability, and we cannot simply discard noisy data by deleting them as some very interesting insights may be part of them. Efforts are made to increase scalability of outlier detection for effectively exploring anomalies in large data sets [45].
c. **Heterogeneour Naature of Data:** It is the Variety characteristics of Big Data that gather & present data collected from various sources, in different formats and are thus essentially heterogeneous in nature. These heterogeneous data in different formats e.g. unstructured, text, audio and video data formats [46] poses challenges to ML algorithms vis–vis their learning rate. We cannot treat all the features of a data set equally important and concatenate them into one as it won't provide an optimal learning outcome and optimal performance. Big Data is seen as an opportunity to learn from multiple views in parallel and then learn the importance of feature views w.r.t. the task to be accomplished. Thus, it will be robust to the data outliers to address optimization and data convergence issues [47]. The heterogeneous mixture data i.e. the collection and storage of mixed data based on different patterns or rules can be challenging in analysis of large scale data. The solution to deal with such data has been proposed by the authors [36] where they make a mention of 'heterogeneous mixture learning' – an advanced form of analysis technology developed by NEC.

d. **Discretization of Data:** It is the process of translating the quantitative data into qualitative data resulting in a non-overlapping division of continuous domain. Decision Trees and Naive Bayes are the examples of some ML algorithms which can only deal with discrete data. Attribute discretization leads to categorization of data which are effective for learning task. However, when dealing with Big Data such traditional approaches are not efficient. The solution is parallelization of standard discretization methods by developing a distributed version of the Entropy Minimization Discretizer based on Minimum Description Length Principle in big data platforms, boosting performance as well as accuracy [48]. Another solution is where the data is first sorted based on the values of numerical attributes and then split into fragments of original class attributes [49].

e. **Data Labelling:** Annotations are important in data understanding but the process is quite tedious as data increases in size/dimension. Alternative methods have been proposed for data labelling when dealing with Big Data e.g. Online Crowd-generated repositories which can serve as a source for free annotated training data [50]. Probabilistic program induction is another approach to address human-level concept learning. User-specific context is another issue that must be addressed properly, otherwise it will result in diminished performance.

f. **Imbalance of Data:** Traditional methods such as stratified random sampling methods can be time consuming and also cannot efficiently support user-specified data set for value-based sampling. They fail to address Big Data and the solution is parallel data sampling, which are based on multiple distributed index files.

g. **Feature Representation and Feature Selection:** The way the data is represented or features are selected (prominent feature identification) affects the performance of ML algorithms [41]. Current algorithms for the above purpose are not sufficiently equipped to handle Big Data. Different solutions such as distributed feature selection, a low rank matrix approximation, representation learning concept, adaptive feature scaling scheme for ultra-high dimensional feature selection, spectral graph theory based framework, fuzzy clustering are proposed over the years and it is still an active area of research. Deep Neural Network based auto encoding has proven effective in learning video, audio and textual features.

Even prior to the advent of Big Data developing scalable ML algorithms for handling large datasets have been an active research area with different researchers working & proposing newer algorithms over a time period. Now, it has gained new impetus & significance as a result of ever increasing challenges being posed by Big Data. The algorithms scalability was mainly aimed at improving performance efficiency in terms of bettering time complexity and space complexity.

State-of-the-art ML algorithms for Big Data Processing focuses on parallelism by exploiting data geometry in the input and/or algorithm/model space. Parallelism may further be classified into Data Parallelism (e.g. MapReduce, Distributed Graph) and model/parameter parallelism (e.g. multi-threading, MPI /OpenMP). Non-parallelism ML algorithms aim to incorporate much faster optimization methods which can deal with big data without any parallelism [40]. Most of the existing work on Big Data Processing using ML focuses on the first three Vs namely Volume, Velocity and Variety aspects, but there is a need to focus on other Vs as well viz; Veracity and Value aspects.

4.2 Trends and Open Issues in Big Data Processing Using Machine Learning

As we understand now that ML based methods and their applications are an integral part of Big Data Processing, it is a hot research area with many new developments happening in this direction. Although research in ML based application development has achieved significant results boosting deriving meaningful insights from Big Data, much more is yet to be accomplished, in this important domain. Qiu et al. [25] describes following future trends from different perspectives in ML based applications for Big Data Processing.

1. **Data Meaning Perspective:** It implies as to how to make ML more intelligent to achieve context-awareness.
2. **Pattern Training Perspective:** It implies how to avoid the overfitting during the process of training patterns.
3. **Technique Integration Perspective:** It deals with integrating other related techniques with ML for Big Data Processing. Developing a composite, integrated and seamless platform for Big Data Processing have a great research potential.
4. **Privacy & Security Perspective:** It provides a research direction for ensuring security and privacy in Big Data Processing using ML techniques.
5. **Realization and Application Perspective:** How and where one must apply ML research in Big Data to gain optimal results. Applying and utilizing the developed ML techniques to real world problems carries huge potential as research area.

5 Conclusion

Big data are now quickly expanding in all science and engineering domains. Learning and gaining newer insights from these massive data brings tremendous opportunities for business houses. Traditional Machine Learning methods for Big Data Processing are not efficient & are not scalable to meet up the high Volume, Velocity, Variety, Veracity and Value (the famous 5 Vs of Big Data), hence ML needs to reinvent itself for big data processing. Machine Learning algorithm based methods are inseparable part of Big Data Processing to gather new unforeseen insights, discover new knowledge and improve efficiency. The amalgamation of ML and Big Data will augur well for the future of data driven industry.

The paper is targeted at providing both the current practices & future research directions in the domain of Big Data Processing using Machine Learning techniques. It is envisaged that the academia and industry will focus on the Veracity & Value aspects in the coming times to improve the processing capabilities, trust management and covering all the important aspects of Big Data. The research scientists, data scientists, analysts and big data practitioners must collaborate towards establishing more efficient Big Data Processing using ML standards and exploring new domains in future.

References

1. Sandryhaila, A., Moura, J.M.: Big data analysis with signal processing on graphs: representation and processing of massive data sets with irregular structure. IEEE Signal Process. Mag. **31**(5), 80–90 (2014)
2. Gantz, J., Reinsel, D.: Extracting value from chaos technical report white paper. International Data Corporation (IDC) Sponsored by EMC Corporation (2011)
3. Gantz, J., Reinsel, D.: The Digital Universe Decade - Are You Ready?. Basic Books, New York (2010)
4. Press, G.: 6 predictions for the $125 billion big data analytics market in 2015 (2014)
5. The evolution of big data, and where we're headed — wired. https://www.wired.com/insights/2014/03/evolution-big-data-headed/. Accessed 10 June 2017
6. Inc., T.P.F.S.G.: The evolution of big data. https://content.pncmc.com/live/pnc/corporate/pncideas/articles/CIB_ENT_PDF_0815-066-196209-CIB_FPS_BigData_rev1.pdf. Accessed 10 June 2017
7. Hype cycle for big data (2014). https://www.gartner.com/doc/2814517/hype-cycle-big-data-. Accessed 10 June 2017
8. Hype cycle - wikipedia. https://en.wikipedia.org/wiki/Hype_cycle. Accessed 10 June 2017
9. Gartner hype cycle for emerging technologies: AI, AR/VR, digital platforms — what's the big data? https://whatsthebigdata.com/2017/08/16/2017-gartner-hype-cycle-for-emerging-technologies-ai-arvr-digital-platforms/. Accessed 10 June 2017
10. What is the difference between artificial intelligence and machine learning? https://www.forbes.com/sites/bernardmarr/2016/12/06/what-is-the-difference-between-artificial-intelligence-and-machine-learning/2/#1f240102483d. Accessed 10 June 2017
11. Jordan, M.I., Mitchell, T.M.: Machine learning: trends, perspectives, and prospects. Science **349**(6245), 255–260 (2015)
12. Fayyad, U., Piatetsky-Shapiro, G., Smyth, P.: From data mining to knowledge discovery in databases. AI Mag. **17**(3), 37 (1996)
13. Ingersoll, G.: Introducing apache mahout. IBM developer Works Technical Library (2009)
14. Mikut, R., Reischl, M.: Data mining tools. Wiley Interdisc. Rev. Data Mining Knowl. Discov. **1**(5), 431–443 (2011)
15. Chen, H., Chiang, R.H., Storey, V.C.: Business intelligence and analytics: From big data to big impact. MIS Q. **36**(4), 1165–1188 (2012)
16. Dietrich, D., Heller, B., Yang, B.: Data Science & Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data. Wiley, Hoboken (2015)

17. Chopra, A., Madan, S.: Big data: a trouble or a real solution? Int. J. Comput. Sci. Issues **12**(2), 221 (2015)
18. Twardowski, B., Ryzko, D.: Multi-agent architecture for real-time big data processing. In: 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), vol. 3, pp. 333–337. IEEE (2014)
19. Amatriain, X.: Mining large streams of user data for personalized recommendations. ACM SIGKDD Explor. Newsl. **14**(2), 37–48 (2013)
20. Richter, A.N., Khoshgoftaar, T.M., Landset, S., Hasanin, T.: A multi-dimensional comparison of toolkits for machine learning with big data. In: 2015 IEEE International Conference on Information Reuse and Integration (IRI), pp. 1–8. IEEE (2015)
21. Wold, S., Esbensen, K., Geladi, P.: Principal component analysis. Chemom. Intell. Lab. Syst. **2**(1–3), 37–52 (1987)
22. Agneeswaran, V.S., et al.: Big-data-theoretical, engineering and analytics perspective. In: BDA, pp. 8–15. Springer (2012)
23. Lehmann, D., Fekete, D., Vossen, G.: Technology selection for big data and analytical applications. Technical report, Working Papers, ERCIS-European Research Center for Information Systems (2016)
24. A short history of machine learning - every manager should read. https://www.forbes.com/sites/bernardmarr/2016/02/19/a-short-history-of-machine-learning-every-manager-should-read/2/#28d56abd6b1b. Accessed 10 June 2017
25. Qiu, J., Wu, Q., Ding, G., Xu, Y., Feng, S.: A survey of machine learning for big data processing. EURASIP J. Adv. Signal Process. **2016**(1), 67 (2016)
26. Zheng, J., Shen, F., Fan, H., Zhao, J.: An online incremental learning support vector machine for large-scale data. Neural Comput. Appl. **22**(5), 1023–1035 (2013)
27. Mitchell, T.M., et al.: Machine Learning. WCB/McGraw-Hill, USA (1997)
28. Ghosh, C., Cordeiro, C., Agrawal, D.P., Rao, M.B.: Markov chain existence and Hidden Markov models in spectrum sensing. In: 2009 IEEE International Conference on Pervasive Computing and Communications, PerCom 2009, pp. 1–6. IEEE (2009)
29. Yue, K., Fang, Q., Wang, X., Li, J., Liu, W.: A parallel and incremental approach for data-intensive learning of Bayesian networks. IEEE Trans. Cybern. **45**(12), 2890–2904 (2015)
30. Dong, X., Li, Y., Wu, C., Cai, Y.: A learner based on neural network for cognitive radio. In: 2010 12th IEEE International Conference on Communication Technology (ICCT), pp. 893–896. IEEE (2010)
31. Safatly, L., Bkassiny, M., Al-Husseini, M., El-Hajj, A.: Cognitive radio transceivers: RF, spectrum sensing, and learning algorithms review. Int. J. Antennas Propag. **2014**, 21 (2014)
32. Bkassiny, M., Jayaweera, S.K., Li, Y.: Multidimensional dirichlet process-based non-parametric signal classification for autonomous self-learning cognitive radios. IEEE Trans. Wirel. Commun. **12**(11), 5413–5423 (2013)
33. Das, T.K., Gosavi, A., Mahadevan, S., Marchalleck, N.: Solving semi-markov decision problems using average reward reinforcement learning. Manag. Sci. **45**(4), 560–574 (1999)
34. Sutton, R.S.: Learning to predict by the methods of temporal differences. Mach. Learn. **3**(1), 9–44 (1988)
35. Najafabadi, M.M., Villanustre, F., Khoshgoftaar, T.M., Seliya, N., Wald, R., Muharemagic, E.: Deep learning applications and challenges in big data analytics. J. Big Data **2**(1), 1–21 (2015)

36. Ryohei, F., Satoshi, M.: The most advanced data mining of the big data era. NEC Tech. J. **7**(2), 91–95 (2012)
37. Jones, N.: The learning machines. Nature **505**(7482), 146 (2014)
38. Langford, J.: Tutorial on practical prediction theory for classification. J. Mach. Learn. Res. **6**(Mar), 273–306 (2005)
39. Bekkerman, R., El-Yaniv, R., Tishby, N., Winter, Y.: Distributional word clusters vs. words for text categorization. J. Mach. Learn. Res. **3**(Mar), 1183–1208 (2003)
40. Zhou, L., Pan, S., Wang, J., Vasilakos, A.V.: Machine learning on big data: opportunities and challenges. Neurocomputing **237**, 350–361 (2017)
41. Bengio, Y., Courville, A., Vincent, P.: Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. **35**(8), 1798–1828 (2013)
42. Chen, Q., Zobel, J., Verspoor, K.: Evaluation of a machine learning duplicate detection method for bioinformatics databases. In: Proceedings of the ACM Ninth International Workshop on Data and Text Mining in Biomedical Informatics, pp. 4–12. ACM (2015)
43. Rakthanmanon, T., Campana, B., Mueen, A., Batista, G., Westover, B., Zhu, Q., Zakaria, J., Keogh, E.: Addressing big data time series: mining trillions of time series subsequences under dynamic time warping. ACM Trans. Knowl. Discov. Data **7**(3), 10 (2013)
44. García, S., Ramírez-Gallego, S., Luengo, J., Benítez, J.M., Herrera, F.: Big data preprocessing: methods and prospects. Big Data Anal. **1**(1), 9 (2016)
45. Cao, L., Wei, M., Yang, D., Rundensteiner, E.A.: Online outlier exploration over large datasets. In: Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 89–98. ACM (2015)
46. Gandomi, A., Haider, M.: Beyond the hype: big data concepts, methods, and analytics. Int. J. Inf. Manag. **35**(2), 137–144 (2015)
47. Cai, X., Nie, F., Huang, H.: Multi-view k-means clustering on big data. In: IJCAI, pp. 2598–2604 (2013)
48. Ramírez-Gallego, S., García, S., Mouriño-Talín, H., Martínez-Rego, D., Bolón-Canedo, V., Alonso-Betanzos, A., Benítez, J.M., Herrera, F.: Data discretization: taxonomy and big data challenge. Wiley Interdisc. Rev. Data Min. Knowl. Discov. **6**(1), 5–21 (2016)
49. Zhang, Y., Cheung, Y.M.: Discretizing numerical attributes in decision tree for big data analysis. In: 2014 IEEE International Conference on Data Mining Workshop (ICDMW), pp. 1150–1157. IEEE (2014)
50. Nguyen-Dinh, L.V., Rossi, M., Blanke, U., Tröster, G.: Combining crowd-generated media and personal data: semi-supervised learning for context recognition. In: Proceedings of the 1st ACM International Workshop on Personal Data Meets Distributed Multimedia, pp. 35–38. ACM (2013)

Harnessing the Power of Big Data in Science

Nitu Bhatnagar(✉)

Department of Chemistry, Manipal University Jaipur, Jaipur, Rajasthan, India
nitu.bhatnagar@jaipur.manipal.edu

Abstract. Big data had been a buzz word till last decade and has shown a tremendous development and application in the computing domain and more so in data storage, data analysis and data sharing. But, at the same time, it has attracted intensive attention in basic disciplines of science like physics, chemistry, biology etc. As research in these areas is increasingly becoming digitized, there is a need to quantify these data so that they can be managed and shared throughout the scientific community. Thus, research in these areas is all set to harness the power of data science and big data analytics and hence there are tremendous opportunities available in this domain.

Keywords: Big data · Science · Physics · Chemistry · Biology

1 Introduction

Big Data has become the driver for innovation in recent years, from strategy decision making to scientific computing [1]. It has shown a tremendous development and application in the computing domain and more so in data storage, data analysis and data sharing. But, at the same time, it has attracted intensive attention in basic disciplines of science like physics, chemistry, biology etc. Research in these areas generate large amount of data which needs to be stored for future use by scientific community anywhere in the world. As research in these areas is increasingly becoming digitized, there is a need to quantify these data so that they can be managed and shared throughout the scientific community. Thus, scientists have felt a need to tackle challenges in storing, handling and interpreting large amount of information. Big data is one such upcoming field which seems to offer solution to all these challenges. The applications and relevance of Big Data in science is also increasing day by day, we are witnessing expansion of Big Data applications as a result of large scale scientific experiments being carried out all over the world generating humongous amount of experimental data. These experimental data was not known in the past is providing many valuable insights to the scientists & experimenters. Thus, Big Data applications and requirement are growing in scientific domain as well. These analytical studies have been made possible by the technological advancements in storage, computing and data processing capabilities of modern day devices.

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 479–485, 2018.
https://doi.org/10.1007/978-3-319-74690-6_47

The following sections discuss the recent advancement, need and applications of Big Data based processing & research in different domains of basic sciences.

2 Big Data Applications in Physics

In physics, big data has shown its presence by the successful storage and analysis of massive data for crustal movement which in turn can provide reliable data and a theoretical basis for earthquake prediction, resource exploration, weather forecasts etc. Another example is that of the ATLAS Experiment at the Large Hadron Collider (LHC) accelerator at CERN in Geneva which has two beams of 100 billion subatomic particles, collided at high speeds in the hope of finding evidence of new physics [2]. It generates about 40 TB of raw data per second which is filtered down to 1 GB per second. At the same time, the raw data is further converted into a data that can be used for physics analysis, resulting in various separate sets of data pose a major challenge. All these data are being put on the Worldwide LHC Computing Grid (WLCG), consisting of 167 computing sites located in 42 countries and holding over 200 PB (200,000 TB) in 1 billion files [2]. With the quantum computing using qubits instead of bits, quantum physics is expected to have a bigger impact on data in the future. This has enabled the concept of quantum entanglement which is expected to create a way of transporting data with no way of intercepting it, thereby allowing hack-proof communications through large distances [3]. The storage, reduction and analysis of these complex data have been restricted to the people who have been involved with these experiments as discussed above. But, of late there had been pressure on those concerned above to share these complex data with the outside world where other researchers can also get benefitted by utilizing these data for their own purpose. Providing open access to large complex data sets is not that easy as it requires technical effort (in long term curation, data reduction and associated metadata production, data delivery in commonly used data formats, software to read and visualize the data, and associated documentation) as well as an understanding of the needs of the broader research community [4]. So, Physics is one important field of Basic Science which is witnessing huge application of Big Data in modern times. The literature is full of many more similar examples as listed above.

3 Big Data Applications in Chemistry

In Chemistry, big data has been widely used in almost all its sub-domains be it inorganic chemistry, organic chemistry and analytical chemistry. It helps in further development and application in the calculation of chemical output [1]. Recently, there has been a remarkable increase in the amount of available compound activity and biomedical data [5–7] which has made the use of big data in chemistry more frequent. “Big Data” in chemistry refers to considerably larger databases than commonly used ones (in orders of magnitude) [8]. This has been possible due to the emergence of new experimental techniques such as high throughput screening, parallel synthesis etc. [6,9] or as access to chemical information as a result of automatic data mining (e.g., patents, literature etc.) [10,11]. Thus, mining large scale data in chemistry has become an important problem for the future development of the chemical industry including pharmaceutical, agrochemical, biotechnological, fragrances, to name a few [12].

As per a report published online by Tetko et al., publicly available databases such as PubChem [6], BindingDB [9], and ChEMBL [7] represent examples of large public domain repositories of compound activity data. Table 1 below lists some popular publicly available large chemical databases. PubChem was originally started as a central repository of High Throughput (HTS) screening experiments for the National Institute of Health's (USA) Molecular Libraries Program but also incorporates data from other repositories like ChEMBL and BindingDB. While ChEMBL and BindingDB contain manually extracted data from tens of thousands of articles, commercial databases, such as SciFinder, GOSTAR and Reaxys contain large amount of data collected from publications and patent data. Apart from these public and commercially available repositories, industry has also produced large private collections. For example, more than 150 M data points are available as part of AstraZeneca International Bioscience Information System (AZ IBIS) just for experiments performed before 2008 [12]. Accumulated chemical patents represent another rich resource for chemical information. The data quality in databases can significantly vary depending on data source, data acquisition procedures and curation efforts.

Large scale text mining has been done on patent corpus to extract useful information. IBM has contributed chemical structures from pre 2000 patents in PubChem [13]. SureChEMBL database [14] was launched in 2014 providing

Table 1. Data repositories [12]

Database	Unique compounds	Experimental		Main data types
		Facts	Data types	
ChEMBL v. 21 [7]	1,592,191	13,968,617	1,212,831	PubChem HTS assays and data mined from literature
BindingDB [9]	529,618	1,207,821	6,265	Experimental protein-small molecule interaction data
PubChem [6]	>60M	>157M	>1M	Bioactivity data from HTS assays
Reaxys [15]	>74M	>500M	-	Literature mined property, activity and reaction data
SciFinder (CAS) [16]	>111M	>80M	-	Experimental properties, 13C and 1H NMR spectra, reaction data
GOSTAR [17]	>3M	>24M	>5k	Target-linked data from patents and articles
AZ IBIS [11]	-	>150M	-	AZ in-house SAR data points
OCHEM [18]	>600k	>1.2M	>400	Mainly ADMET data collected from literature

the wealth of knowledge hidden in patent documents and currently contains 17 million compounds extracted from 14 million patent documents.

4 Big Data Applications in Biology and Health Care

In the field of biology and health care, great breakthroughs have been made in acquiring and mapping sequences of human genome, the analysis of disease gene sequences, and the targeted disease treatment. The main advantage of applying Big Data Analytics in health care is that it saves time, while improving the efficacy of treating a particular disease condition.

In bioinformatics, high throughput experiments facilitate the research of new genomewide association studies of diseases, and with clinical informatics, the clinical field benefits from the vast amount of collected patient data for making intelligent decisions. Imaging informatics is now more rapidly integrated with cloud platforms to share medical image data and workflows, and public health informatics leverages big data techniques for predicting and monitoring infectious disease outbreaks, such as Ebola [19]. In physics and biology, big data is measured in 'petabytes' (1 PB = 1015 bytes) and cloud computing has become an indispensable part of big data storage and analysis [20]. Mining the data can help analyze what treatments are most effective for particular conditions, identify patterns related to drug side effects or hospital readmissions, and gain other important information that can help patients and reduce cost [21].

Under the enormous pressure of developing new drug with more restrained R&D budget, recent years have seen large pharma companies increasingly exploring the so called "open innovation" model for drug discovery research. The collaboration between academics and pharmaceutical industry in terms of compound, data sharing has been largely increased [22]. The examples include AstraZeneca-Sanger Drug Combination prediction challenge to develop better algorithms for treatment of cancer [23].

Drug discovery is another area where Big data has brought about a considerable change in the field of drug discovery to the extent that it is being felt that there is an urgent need to explore new training concepts for discovery scientists. In contrast to this, big data is still in its growing stage in the field of medicinal chemistry which is another pillar of drug discovery. The area of medicinal chemistry is one such scientific discipline where big data is beginning to emerge and provide new opportunities and challenges, as exemplified by the computational study of biological activities of drugs and other compounds from medicinal chemistry [20]. For example, the ability of many drugs to specifically interact with multiple targets, termed promiscuity, forms the molecular basis of polypharmacology, a hot topic in drug discovery. Compound promiscuity analysis is an area that is much influenced by big data phenomena.

Clearly, the variety of data associated with cells and organisms is principally much larger than of data associated with chemical compounds – and so are the ensuing data volumes that can be generated. However, although big data trends are only beginning to emerge in medicinal chemistry [24,25] it is evident that this field will also be increasingly influenced by big data issues. For example, proprietary medicinal chemistry projects in the pharmaceutical industry will inevitably need to take compound activity data into consideration that is rapidly accumulating in the public domain [24]. Merging internal and external data and viewing chemical optimization of compound series in an overarching context represents a departure from the long established operating culture of medicinal chemistry and presents new challenges to practicing chemists. However, the opportunities provided by extracting knowledge from rapidly growing amounts of compounds and publicly available activity data cannot be disregarded.

One major challenge that one can foresee is that big data in medicinal chemistry primarily focuses on experimental data, which typically require computational analysis – but not on computationally generated data. If thousands of chemical descriptors and properties can be calculated for any given compound, it is evident that the amount of compound associated data can be further increased by orders of magnitude through computational chemistry. However, such 'theoretical' big data represents another category to which big data criteria considered herein only vaguely apply and the utility of which – and relevance for the practice of medicinal chemistry – might also be questioned [20].

5 Conculsions

Complex problems not only in areas of physics, chemistry and biology, but also interdisciplinary areas of science such as drug discovery, genome analysis, and medicinal chemistry commonly require collection and analysis of a vast amount of data. All these areas showcase one or more big data characteristics of volume, velocity and veracity. These big data characteristics present computational and analytical challenges that need to be overcome in order to make scientific discoveries. Harnessing powerful computers and numerous tools for data analysis is crucial in analysing such vast data available in any area of science. There has to be specialized and skilled persons dealing with these data and tools. Another important challenge which is to be addressed in Big Data analytics is that data available for mining should be in the usable form for use by scientific community.

References

1. Dehmer, M., Emmert-Streib, F., Pickl, S., Holzinger, A.: Big Data of Complex Networks. CRC Press, Boca Raton (2016)
2. How Big Data Advances Physics. https://www.elsevier.com/connect/how-big-data-advances-physics. Accessed 10 July 2017
3. Quantum Physics and the Big Data Question | Articles | Chief Data Officer | Innovation Enterprise. https://channels.theinnovationenterprise.com/articles/quantum-physics-and-the-big-data-question. Accessed 10 July 2017
4. Abstract: Big Science, Big Data, Big Challenges: Data from Large-Scale Physics Experiments (2014 AAAS Annual Meeting, 13–17 February 2014). https://aaas.confex.com/aaas/2014/webprogram/Paper10566.html. Accessed 25 Oct 2017

5. Chen, B., Butte, A.: Leveraging big data to transform target selection and drug discovery. Clin. Pharmacol. Therapeutics **99**(3), 285–297 (2016)
6. Kim, S., Thiessen, P.A., Bolton, E.E., Chen, J., Fu, G., Gindulyte, A., Han, L., He, J., He, S., Shoemaker, B.A., et al.: Pubchem substance and compound databases. Nucleic Acids Res. **44**(D1), D1202–D1213 (2015)
7. Papadatos, G., Gaulton, A., Hersey, A., Overington, J.P.: Activity, assay and target data curation and quality in the chembl database. J. Comput. Aided Mol. Des. **29**(9), 885–896 (2015)
8. Tetko, I.V., Lowe, D.M., Williams, A.J.: The development of models to predict melting and pyrolysis point data associated with several hundred thousand compounds mined from patents. J. Cheminform. **8**(1), 2 (2016)
9. Gilson, M.K., Liu, T., Baitaluk, M., Nicola, G., Hwang, L., Chong, J.: BindingDB in 2015: a public database for medicinal chemistry, computational chemistry and systems pharmacology. Nucleic Acids Res. **44**(D1), D1045–D1053 (2016)
10. Schneider, N., Lowe, D.M., Sayle, R.A., Tarselli, M.A., Landrum, G.A.: Big data from pharmaceutical patents: a computational analysis of medicinal chemists' bread and butter. J. Medicinal Chem. **59**(9), 4385–4402 (2016)
11. Muresan, S., Petrov, P., Southan, C., Kjellberg, M.J., Kogej, T., Tyrchan, C., Varkonyi, P., Xie, P.H.: Making every SAR point count: the development of chemistry connect for the large-scale integration of structure and bioactivity data. Drug Discov. Today **16**(23), 1019–1030 (2011)
12. Tetko, I.V., Engkvist, O., Koch, U., Reymond, J.L., Chen, H.: Bigchem: challenges and opportunities for big data analysis in chemistry. Mol. Inform. **35**(11–12), 615–621 (2016)
13. IBM Contributes Data to the National Institutes of Health to Speed Drug Discovery and Cancer. http://www.prnewswire.com/news-releases/ibm-contributes-data-to-the-national-institutes-of-health-to-speed-drug-discoveryand-cancer-research-innovation-135275888.html. Accessed 10 July 2017
14. Papadatos, G., Davies, M., Dedman, N., Chambers, J., Gaulton, A., Siddle, J., Koks, R., Irvine, S.A., Pettersson, J., Goncharoff, N., et al.: Surechembl: a large-scale, chemically annotated patent document database. Nucleic Acids Res. **44**(D1), D1220–D1228 (2015)
15. Chemistry Data and Literature - Reaxys | Elsevier. https://www.elsevier.com/solutions/reaxys. Accessed 10 July 2017
16. Scifinder - A CAS Solution. http://www.cas.org/products/scifinder. Accessed 10 July 2017
17. Sarma, J.: Gostar: GVK bio online structure activity relationship database: data and its utility. In: Abstracts of Papers of the American Chemical Society, vol. 238. American Chemical Society, Washington, DC (2009)
18. Sushko, I., Novotarskyi, S., Körner, R., Pandey, A.K., Rupp, M., Teetz, W., Brandmaier, S., Abdelaziz, A., Prokopenko, V.V., Tanchuk, V.Y., et al.: Online chemical modeling environment (OCHEM): web platform for data storage, model development and publishing of chemical information. J. Comput. Aided Mol. Des. **25**(6), 533–554 (2011)
19. Luo, J., Wu, M., Gopukumar, D., Zhao, Y.: Big data application in biomedical research and health care: a literature review. Biomed. Inform. Insights **8**, 1 (2016)
20. Hu, Y., Bajorath, J.: Entering the 'big data' era in medicinal chemistry: molecular promiscuity analysis revisited. Future Sci. OA **3**(2), FSO179 (2017)
21. Kayyali, B., Knott, D., Van Kuiken, S.: The Big-Data Revolution in US Health Care: Accelerating Value and Innovation, vol. 2, no. 8, pp. 1–13. Mc Kinsey & Company (2013)

22. Allarakhia, M., Director, B.N.: Pfizer's centers for therapeutic innovation (2011)
23. Astrazeneca-Sanger Drug Combination Prediction Dream Challenge. https://www.synapse.org/#!Synapse:syn4231880/wiki/235645. Accessed 10 July 2017
24. Hu, Y., Bajorath, J.: Learning from 'big data': compounds and targets. Drug Discov. Today **19**(4), 357–60 (2014)
25. Lusher, S.J., McGuire, R., van Schaik, R.C., Nicholson, C.D., de Vlieg, J.: Data-driven medicinal chemistry in the era of big data. Drug Discov. Today **19**(7), 859–868 (2014)

Building Online Social Network Dataset for Arabic Text Classification

Ahmed Omar[1](✉), Tarek M. Mahmoud[1,2], and Tarek Abd-El-Hafeez[1]

[1] Computer Science Department, Faculty of Science, Minia University,
EL-Minia, Egypt
{Ahmed. omar, d. tarek, tarek}@mu. edu. eg
[2] Canadian International College (CIC), Cairo, Egypt

Abstract. Social networking sites have spread widely in recent years, and through them, a large amount of data is shared in all its forms: text, photo, voice, and video. It also allows communication with users through different forms such as chat, comments, and Posts, and the most exchanged content is in the form of text data. These results a large volume of data displayed to each user. This encouraged and attracted the attention of researchers to make an effort to analyze and work on this large amount of data available for free on the online social networks, most efforts focus on Twitter and English data. Building dataset is the most time-consuming and the most important part of the text classification process. Despite the increase in the number of Arabic users and the increase in Arabic content on online social Networks (OSN), there is a scarcity in Arabic datasets collected from social networks for text classification purpose. So In this paper, Arabic social dataset was built to be used in text classification purpose. our dataset was gathered from Facebook, it consists of 25,000 posts were collected from different Facebook pages and were classified into ten categories, politics, economics, sport, religion, technology, TV, ads, foods, health, and porno. The dataset was assessed to ten Arabic local speakers and Facebook users to evaluate the validity of the dataset made. We used a RapidMiner tool to evaluate and compute the performance of our dataset. We obtained a classification accuracy of 95.12%.

Keywords: Text mining · Text classification · Online social network
Arabic text · Arabic dataset

1 Introduction

Research in social media has become a point of interest from many researchers because of the increasing field of online social networks in most platforms. Social Networks are nowadays the most familiar interactive media to communicate, share, and publish an unlimited amount of human life information. Communications mean the exchange of particular types of content, including text, photo, audio, and video data. Online Social Networks supply very little support to prevent unwanted data on user timeline. Sometimes the shared information may be vulgar or not wanted and it is inevitable to see it. Facebook, for example, gives users the ability to declare who is allowed to add

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 486–495, 2018.
https://doi.org/10.1007/978-3-319-74690-6_48

data to their walls. (i.e., friends, friends of friends, or defined groups of friends). In Facebook, no data checking for the contents happen and hence it is highly likely that offensive content gets posted without unchecking or filter no matter of the users [1].

Most of the exchanged data over the Social Networks are in the text format. Text mining is a technique made together with data mining, machine learning, and information retrieval. Text mining may also point out as text data analysis or data mining in which significant information can be retrieved from the text. To make text preprocessing or prerequisites, the phases are parsing, tokenization, normalization, etc. [2]. Text classification techniques will be used for automatically labeling a set of categories based on contents of each text data. The classification will be one of the following categories (politics, economics, sport, religion, technology, TV, ads, foods, health, and porno).

Online Social Networks (OSN) are used by different languages' speakers. It is not only used by English speakers. There are many users on OSN use other languages than English e.g. Arabic. Arabic is fourth one of the top ten languages on the internet in June 2017 [3]. The need and attention in classifying Arabic texts have increased recently, due to a lot of reasons: The Arabic language is very rich with contents, there are about 184 million Arab Internet users and a large percentage of them cannot read English [3]. In addition to, the online Arabic contents have grown quickly in the last decade, exceeding 3% of the entire online contents and is ranked the eighth in the whole internet content [4]. However, there is lack of language resources and text processing techniques for the Arabic language [5].

This paper presents a dataset collected from Arabic Facebook pages, the dataset contains 25,000 posts, collected automatic and manually labeled to 10 categories, and then we apply some text processing techniques which include, removing non-Arabic letters, removing word suffix and prefix, normalization, and transformation. The labeling phase includes two stages, in the first, we labeled each post to a specific class, then we ask 10 Facebook users who are Arabic native speakers to labeled the posts, and according to users' feedbacks, some posts classification are changed.

The next sections are organized as follows: Sect. 2 contains related works review. In Sect. 3 the data collection methodology is viewed. Section 4 contains results and evaluation followed by Sect. 5 which contains the conclusion.

2 Related Work

The dataset building is different in terms of the research purpose, such as Natural Language Processing (NLP) and Text Mining. The dataset differs also in the collect sources i.e. websites, social networks, news pages, and blogs. Also, datasets vary in size and language. Some datasets are available for free and some of them are available commercially. In recent years interest is begun to build datasets from online social networks, because of the large amounts of data available in it. There are no free standard datasets available for the Arabic text classification research, unlike English text classification, so researchers rely on collecting their dataset for each research point [4]. Few research efforts were done for Arabic datasets building.

Al-Kabi et al. [6] collect 4050 comments from social media such as Facebook, YouTube, Twitter, Digg, and Yahoo. These comments were used to build a dataset in

Arabic and English languages, SocialMention and Twendz tools were used to gather comments together with reviews in Arabic and English language. The dataset was classified to three classes only, political news, commercial, and academic. Three classification models were used to evaluate the dataset ((Naïve Bayes, Support Vector Machine (SVM), and K-Nearest Neighbor algorithm (K-NN)), and the conducted results showed that the Naïve Bayes algorithm gave the best results for both SocialMention and Twendz tools with an accuracy of 66.2% and 45.3%, respectively.

Abdul-Mageed et al. [7] created annotated data comprising a four of datasets contain different dialects: the first dataset contains 2798 chat message collected randomly from of an Egyptian room chat session in Maktoob chat, the second dataset contains 3015 Arabic tweets collected from Twitter. The third dataset consists of 3008 sentences, was collected from 30 Talk Pages on Wikipedia. The fourth one comprises 3097 sentences Web forum collected from a larger pool of threaded conversations pertaining to different varieties of Arabic, the topics covered in this forum is religion or politics. The proposed system by Abdul-Mageed et al. gives the best accuracy with the first dataset, which achieves 84.65% in subjectivity classification.

Yin et al. [8] built a dataset for short text classification, the data was collected from two micro blogs and contains five classes, Politics, Economy, Education, Entertainment, and S&T. Semi-supervised learning and SVM were used to improve the accuracy of the classification, and the highest performance of this algorithm in terms of precision and recall was 80.49% and 81.77%, respectively.

Al-Tahrawi and Al-Khatib [4] used Al-Jazeera News dataset to evaluate the performance of polynomial networks classifier. Alj-News dataset was collected from Al-jazeera News Arabic Website. The dataset contains 1500 Arabic documents divided evenly into five classes: Politics, Science Economic, Art, and Sport. And the performance in terms of precision and recall was 90% and 89%, respectively.

Al Mukhaiti et al. [9] built a dataset for Arabic Sentiment Analysis. The data was collected from Facebook, YouTube, Twitter, Keek, and Instagram, and contains 2009 tweets/review. A system developed by Siddiqui et al. [10] was used to evaluate the dataset, the evaluation metrics were precision, recall, and accuracy and the results were 75.9%, 79.8, and 77.7%, respectively.

Alayba et al. [11] built a dataset for Arabic Sentiment Analysis on health services, the dataset contains 2026 tweets collected from twitter using twitter API, three machine learning algorithms were used: Naïve Bayes (NB), Logistic Regression (LR), and SVM, with a change on the size of training set and test set in three phases, The accuracy results were between 85% and 91% and the best classifiers was SVM using linear support vector.

3 Data Collection

The most time-exhaustion and the most important phase of text mining is Data collection [9]. In this paper, we will explain the overview of the dataset development process. This process divided into three phases, data Acquisition phase, data filtering phase, and data labeling phase. Figure 1 depicts the dataset development phases.

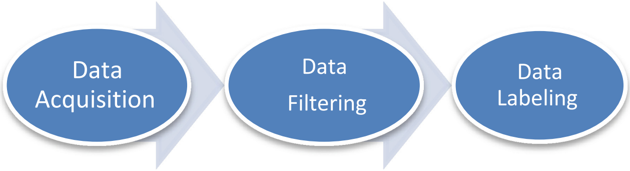


Fig. 1. Dataset development phases

The following steps depict the dataset development phases:

1. Collect data by crawling the Arabic Facebook pages.
2. Filter the data collected in the previous step
   a. Removing URLs
   b. Removing non-Arabic
   c. Removing repeated posts.
3. Manually labeled the filtered posts to one of the ten categories chosen.

We chose ten categories/classes for the dataset, these ten classes cover most of the social network topics, and the classes are politics, economics, sport, religion, technology, TV, ads, foods, health, and porno. Figure 2 shows the process of building the proposed dataset. The algorithm used for building the dataset can be summarized as:

3.1 Data Acquisition Phase

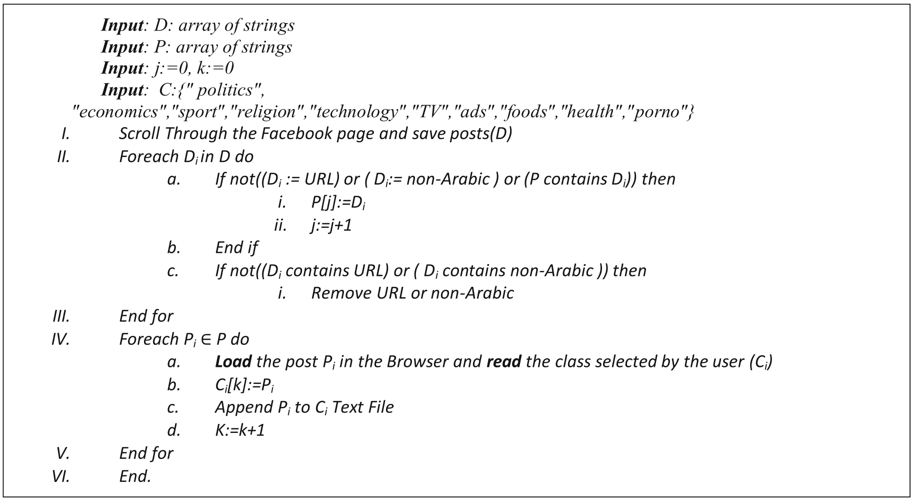
We have collected about 40,000 Arabic Facebook posts. To collect the posts we developed a web browser to collect the data automatically, it has the ability to collect posts, comments, and replies, by automatically scrolling down the Facebook page to show all posts from the page date of creation, then gathering all the posts and save them in a text file.

3.2 Data Filtering Phase

This phase included of removal of the following types of posts, URLs only posts, non-Arabic posts, and repeated posts. Some posts contain only URL(s), this URL is in English letters and does not useful in Arabic dataset. Arabic Facebook pages sometimes share non-Arabic posts, English or Franco Arabic (Arabic spellings with English letters and digits). The last type is repeated posts, different pages may publish the same post, or a page may re-post an old post, the post was added only once. In case that the post contains an Arabic text plus to URLs, digits, non-Arabic letters, or emotions symbols, this post will be filtered, the Arabic text only will be saved, and the remaining parts will be removed. Table 1 depicts some examples.

3.3 Data Labeling Phase

The filtered posts were used in the labeling phase, wherein the filtered posts were labeled as politics, economics, sport, religion, technology, TV, ads, foods, health, or



```
        Input: D: array of strings
        Input: P: array of strings
        Input: j:=0, k:=0
        Input:  C:{"politics",
    "economics","sport","religion","technology","TV","ads","foods","health","porno"}
I.      Scroll Through the Facebook page and save posts(D)
II.     Foreach D_i in D do
            a.  If not((D_i := URL) or ( D_i:= non-Arabic ) or (P contains D_i)) then
                    i.   P[j]:=D_i
                    ii.  j:=j+1
            b.  End if
            c.  If not((D_i contains URL) or ( D_i contains non-Arabic )) then
                    i.   Remove URL or non-Arabic
III.    End for
IV.     Foreach P_i ∈ P do
            a.  Load the post P_i in the Browser and read the class selected by the user (C_i)
            b.  C_i[k]:=P_i
            c.  Append P_i to C_i Text File
            d.  K:=k+1
V.      End for
VI.     End.
```

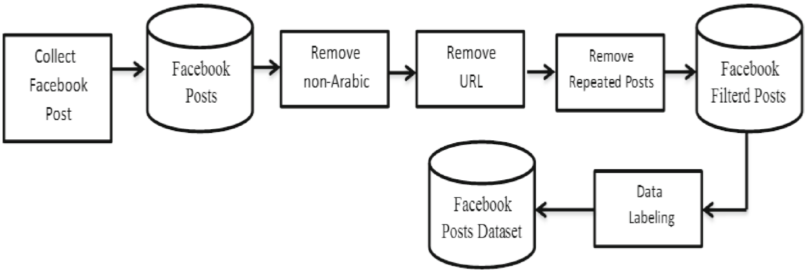


Fig. 2. Arabic dataset building process

Table 1. Filtering examples

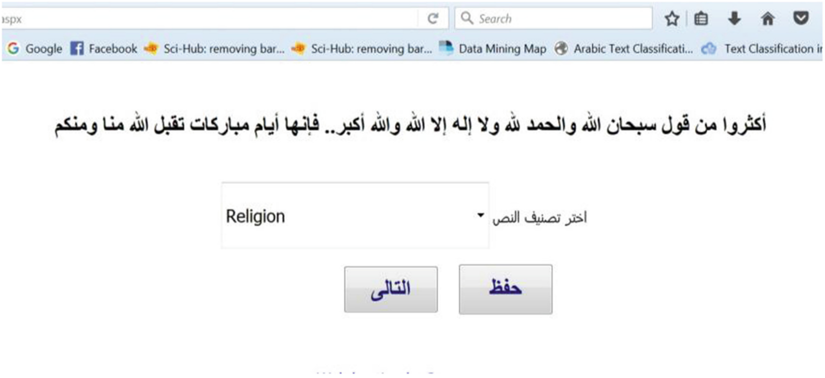
Post	Action-Reason
هدف بوليفيا في شباك التانغو Bolivia’s goal	Filtered-delete non-Arabic
http://onbe.in/1K6M2Up	Removed-Only URL
matsh alahly walzmalk Isa bkrh alsa3h 10	Removed-non-Arabic
جهزي مطبخك بحلول مبتكرة bit.ly/CC_kitchen_utensils	Filtered-delete URL
اختبارات الحساسية لمعرفة سبب المرض☺☺	Filtered-delete emotions
تم ايقاف الورقة المالية EGS02051C018 لمدة نصف ساعة	Filtered-delete non-Arabic

porno. No system accessible for assessing posts for Arabic text classification. So, we divide the labeling phase to two parts, first As a speaker of the Arabic language, I categorized the data collected, by reading and characterize each post to one of the previous ten classes. Table 2 views example of the classified posts.

To assess our labeling made during the first part, the dataset was given to ten Arabic native speakers who additionally confirm the validity of the dataset created. We

Table 2. Labeling phase

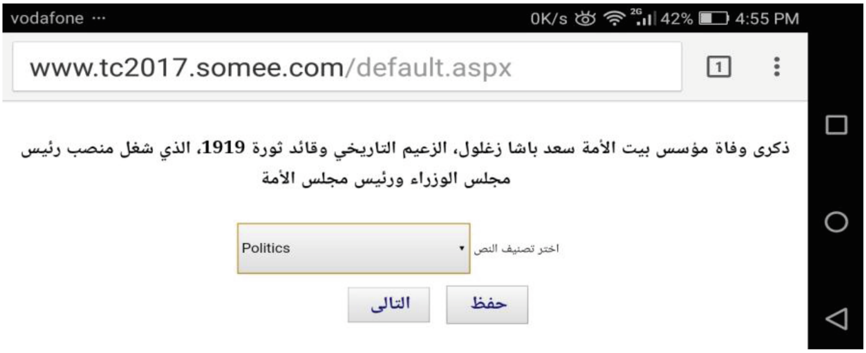
Post	Labeled as
مطلوب سيارات حديثة بسائق أو بدون لفترات طويلة بشيراتون المطار	Ads
النيل للتأجير التمويلى تستهدف طرح أسهمها بالبورصة لتمويل الأنشطة الجديدة	Economics
البيكنج بودر هو سر نجاح التوست أعرفي المقادير بالتفصيل من هنا	Food
علاج السمنة الموضعية بالميزو ثيرابي بعيادة الشفاء هو تقنية آمنة طبية	Health
مشاورات نائب وزير الخارجية المصرى على هامش جنيف	Politics
يارب مع نهاية اليوم نوّر لي قلبي، ويسّر لي حالي، وفرّج عني كربي وهمي	Religion
تعرف علي رسالة حسام حسن للاعبيه قبل مواجعة الزمالك في كأس مصر	Sport
تسريب مواصفات هاتف سامسونج	Technology
محمد هنيدي يبدأ تسجيل حلقات جديدة من سوبر هنيدي	TV
ممكن إنسانة جادة لعلاقة ممتعة دلع وحب	Porno



أكثروا من قول سبحان الله والحمد لله ولا إله إلا الله والله أكبر.. فإنها أيام مباركات تقبل الله منا ومنكم

اختر تصنيف النص Religion

Fig. 3. Webpage screenshot from a PC



www.tc2017.somee.com/default.aspx

ذكرى وفاة مؤسس بيت الأمة سعد باشا زغلول، الزعيم التاريخي وقائد ثورة 1919، الذي شغل منصب رئيس مجلس الوزراء ورئيس مجلس الأمة

اختر تصنيف النص Politics

Fig. 4. Webpage screenshot from a smartphone

built an online web page to help the users to assess the dataset remotely, at any time, from any location and from any device, PC or smartphone. In Figs. 3 and 4 screenshots of the web page from PC and smartphone, are shown respectively.

4 Results and Evaluation

Data Classification is a two steps process: (1) the training (or learning) phase and (2) the test (or evaluation) phase where the actual class of the instance is compared with the predicted class. If the hit rate is acceptable to the analyst, the classifier is accepted as being capable of classifying future instances with unknown class [12].

Our dataset building process contains three phases: 1. Data Acquisition, 2. Data Filtering, and 3. Data Labeling. Table 3 views the total number of posts in each class after phase 3.

Table 3. Labeling dataset phase

Class	No. of posts
Sport	2500
Politics	2500
TV	2500
Technology	2500
Religion	2500
Economic	2500
Food	2500
Porno	2500
Ads	2500
Health	2500
Total	25000

To evaluate our dataset, RapidMiner Studio Professional 7.6 was used to analyze data, RapidMiner is an open-source platform independently used for data mining [13]. RapidMiner is code-free software for designing advanced analysis processes with machine learning, data and text mining and business analytics, and predictive analytics, through its graphical user interface, data mining processes can be easily designed and executed. There are operators for tokenization, stemming, and stop words filtering. RapidMiner provides extensions of data loading, data transformation, data modeling, and data visualization methods. One of the most useful extensions in RapidMiner is the Text Processing package, which includes operators that support text mining. RapidMiner has an important feature, it can process a lot of languages including the Arabic language.

We applying RapidMiner operators: Naive Bayes, k-Nearest Neighbors (k-NN), support vector machine SVM, Performance (for classification) and Apply model (for testing). Evaluation metrics includes weighted mean recall, Weighted mean Precision, Kappa statistic, and Accuracy. The weighted mean recall is the average of recall calculated per class. Weighted Mean Precision is the average of precision obtained per class. Kappa Statistic (the accuracy varies from 0 to 1) measures the approval, of prediction with the true class and it means that the classifier is in total agreement with a random classifier. The accuracy is defined as the ratio of numbers of correctly classified posts to the total number of posts. Recall and Precision are defined as:

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \tag{1}$$

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \tag{2}$$

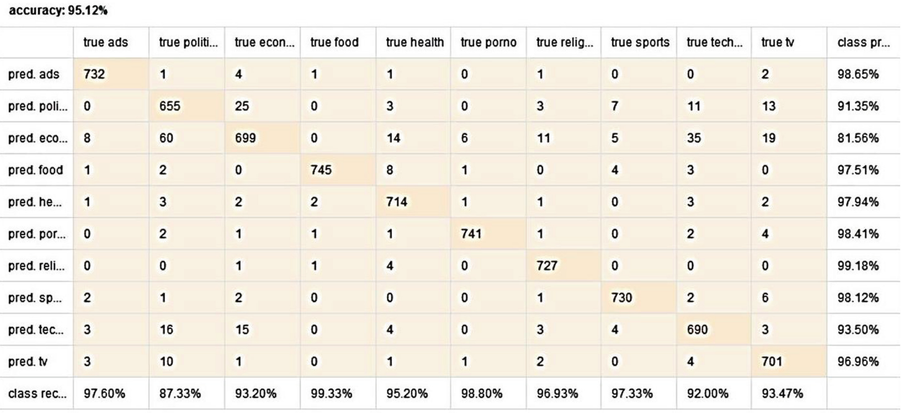
$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{TP} + \text{TN} + \text{FP} + \text{FN}) \tag{3}$$

Where TP, TN, FP, and FN refer to: Truly Positive, Truly Negative, Falsely Positive, and Falsely Negative claims of the classifier respectively.

Table 4 shows the result of testing our dataset on RapidMiner tool. It gives the accuracy of 95.12% when using SVM model, which gives the highest accuracy among used models. Figure 5 depicts the output result from RapidMiner. Comparing our results with other available dataset mentioned in Sect. 2 related work, show that the accuracy of our dataset is high compared to mentioned results as shown in Table 5.

Table 4. Evaluation results

Evaluation metric	Value
Accuracy	95.12%
Weighted mean recall	95.12%
Weighted mean precision	95.32%
Kappa	0.946



accuracy: 95.12%

	true ads	true politi...	true econ...	true food	true health	true porno	true relig...	true sports	true tech...	true tv	class pr...
pred. ads	732	1	4	1	1	0	1	0	0	2	98.65%
pred. poli...	0	655	25	0	3	0	3	7	11	13	91.35%
pred. eco...	8	60	699	0	14	6	11	5	35	19	81.56%
pred. food	1	2	0	745	8	1	0	4	3	0	97.51%
pred. he...	1	3	2	2	714	1	1	0	3	2	97.94%
pred. por...	0	2	1	1	1	741	1	0	2	4	98.41%
pred. reli...	0	0	1	1	4	0	727	0	0	0	99.18%
pred. sp...	2	1	2	0	0	0	1	730	2	6	98.12%
pred. tec...	3	16	15	0	4	0	3	4	690	3	93.50%
pred. tv	3	10	1	0	1	1	2	0	4	701	96.96%
class rec...	97.60%	87.33%	93.20%	99.33%	95.20%	98.80%	96.93%	97.33%	92.00%	93.47%	

Fig. 5. RapidMiner results

Table 5. Comparison of some existing works

Evaluation metric	Proposed dataset	Al-Kabi et al. [6]	Abdul-Mageed et al. [7]	Yin et al. [8]	Al-Tahrawi and Al-Khatib [4]	Al Mukhaiti et al. [9]	Alayba et al. [11]
Accuracy	95.12%	66.2%	84.65%	–	–	77.7%	91%
Recall	95.12%	–	–	81.7%	90%	79.8	–
Precision	95.32%	–	–	80.4%	89%	75.9%	–

5 Conclusions

In recent years, online social networking sites have spread widely, and data is published and shared in large quantities every moment. Social networking sites do not give users ability to filter or categorize content on their walls. Therefore, we have created in this paper a dataset of online Arabic text collected from the Facebook to be used in the text classification process. We chose Arabic because of the paucity of the Arabic dataset available while online Arabic content is increasing and online Arab users are growing. The dataset building process divided into three phases, data Acquisition, data filtering, and data labeling phase. The dataset was collected from Arabic Facebook pages, then in data filtering phase the URLs, non-Arabic, and repeated posts were removed from the dataset. Finally, in the labeling phase, each post was given a label from the ten chosen categories and ten Facebook Arabic users were involved in the labeling process. To evaluate our dataset RapidMiner tool was used and the performance achieved in terms of accuracy was 95.12% with the SVM model. Our dataset will help researchers in the field of short Arabic text processing.

References

1. Bodkhe, R., Ghorpade, T., Jethani, V.: A novel methodology to filter out unwanted messages from OSN user's wall using trust value calculation. In: Proceedings of the Second International Conference on Computer and Communication Technologies, pp. 755–764. Springer (2016)
2. Ghosh, S., Roy, S., Bandyopadhyay, S.: A tutorial review on text mining algorithms. Int. J. Adv. Res. Comput. Commun. Eng. **1**(4), 7 (2012)
3. Internet World Stats: Internet World Users by Language (2017). http://www.internetworldstats.com/stats7.htm. Accessed 13 Sep 2017
4. Al-Tahrawi, M.M., Al-Khatib, S.N.: Arabic text classification using polynomial networks. J. King Saud Univ. Comput. Inf. Sci. **27**(4), 437–449 (2015)
5. Al-Sallab, A., Baly, R., Hajj, H., Shaban, K.B., El-Hajj, W., Badaro, G.: AROMA: a recursive deep learning model for opinion mining in Arabic as a low resource language. ACM Trans. Asian Low-Resour. Lang. Inf. Process. (TALLIP) **16**(4), 25 (2017)
6. Al-Kabi, M., Al-Qudah, N.M., Alsmadi, I., Dabour, M., Wahsheh, H.: Arabic/English sentiment analysis: an empirical study. In: The Fourth International Conference on Information and Communication Systems (ICICS 2013), pp. 23–25 (2013)
7. Abdul-Mageed, M., Diab, M., Kübler, S.: SAMAR: subjectivity and sentiment analysis for Arabic social media. Comput. Speech Lang. **28**(1), 20–37 (2014)

8. Yin, C., Xiang, J., Zhang, H., Wang, J., Yin, Z., Kim, J.-U.: A new SVM method for short text classification based on semi-supervised learning. In: 2015 4th International Conference on Advanced Information Technology and Sensor Application (AITS), pp. 100–103. IEEE (2015)
9. Al Mukhaiti, A.J.S., Siddiqui, S., Shaalan, K.: Dataset built for Arabic sentiment analysis. In: International Conference on Advanced Intelligent Systems and Informatics, pp. 406–416. Springer (2017)
10. Siddiqui, S., Monem, A.A., Shaalan, K.: Sentiment analysis in Arabic. In: International Conference on Applications of Natural Language to Information Systems, pp. 409–414. Springer (2016)
11. Alayba, A.M., Palade, V., England, M., Iqbal, R.: Arabic language sentiment analysis on health services. arXiv preprint arXiv:1702.03197 (2017)
12. Borges, L.C., Marques, V.M., Bernardino, J.: Comparison of data mining techniques and tools for data classification. In: Proceedings of the International C* Conference on Computer Science and Software Engineering, pp. 113–116. ACM (2013)
13. RapidMiner Documentation. https://docs.rapidminer.com/. Accessed 10 Sep 2017

Breast Cancer Detection and Classification Using Thermography: A Review

Abdelhameed Ibrahim(✉), Shaimaa Mohammed, and Hesham Arafat Ali

Computer Engineering and Systems Department, Faculty of Engineering,
Mansoura University, Mansoura, Egypt
afai79@mans.edu.eg

Abstract. Cancer is considered as the leading cause of death among people. The cancer is generated from uncontrolled growth for cells to collect them together to construct tumor. One of these cancer types is breast cancer. Detecting breast cancer, which is the second leading cause of death in women after lung cancer, depends on asymmetry in temperature between breasts. If breast cancer can be detected at an early stage, it can save women life. The thermogram is more proper screening and has lower cost than other types of screening methods like the mammogram, ultrasound, and magnetic resonance imaging depending on a temperature of breast and surrounding area by using a special heat-sensing camera to determine the heat in the region of breasts. To classify healthy and unhealthy cases of breast cancer, methods are divided into image acquisition, preprocessing, segmentation, feature extraction and classification. This paper focuses on reviewing the state-of-the-art methods and techniques of detecting and classifying the breast cancer using thermography images.

Keywords: Thermogram · Computational geometry · Graph theory
Breast cancer

1 Introduction

Cancer effects on a cell by growing it up in an abnormal way and change in their shape. If this growth not controlled in an early stage, death occurred. Reasons for cancer could be internal or external reasons. The breast cancer is the most common cancer in women and if it can be detected at an early stage, it can save women life. Breast cancer is the second type of cancer appears to women especially but in men rarely and occur especially in ducts or tubes depending on family history. Using screening method help to save the life. According to statistics of American Cancer Society in 2017, there are 318,590 new cases appears to both women and men also of breast cancer and nearly 41,070 cases died from breast cancer divided into 40,610 women and 460 men [1,2]. Cancer of breast is considered as the second type of all cancer types and the primary source of death in women it constitutes 30% from all cancerous cases [3,4]. Discovery of cancer in early stages for women achieve more savings in their life [5].

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 496–505, 2018.
https://doi.org/10.1007/978-3-319-74690-6_49

There are more than a method to screen cancer breasts such as a mammogram, ultrasonography, computed tomography, magnetic resonance imaging (MRI), thermography and lately more than one type that can be merged with other to early detect breast cancer [6,7]. The mammogram, a kind of X-ray for the breasts, is considered as the most common screening method for detecting cancer. This technique used to early detect small tumors in women in age 50 to 70. It consists of Film-screen mammography which appeared in 1976, then in 1997 the Full-field digital mammography was presented and finally in 2011 the digital breast tomosynthesis or the 3D mammography was introduced [8]. This technique was observed as the most valuable and active tool in breast cancer detection and reduced the mortality rates. However, it was not useful for younger women than forty years old; type of X-Ray which causes damage in tissue; expensive; consume the time and not preferred in medical cases [9,10]. There were some drawbacks to this method and to solve it new screening methods appeared.

Mammogram technique uses various types of database such as Mammographic Image Analysis Society (MIAS) Digital Mammogram Database, mini-MIAS, and merging more than one type of database to detect breast cancer. The second screening method is ultrasound, which usually used after abnormal cells appear in mammogram by merging them. Ultrasound technology has some advantage than mammography such as giving a good result in women age less than thirty-five years old, more safe, cheap, fast and gives a higher rate of accuracy in small tumors [11]. This method had more than a type for screening, the most common type of them was B-mode. The database used was breast ultrasound (BUS) database [12]. Although ultrasound detects breast cancer than mammography, ultrasound using is less than it. Ultrasound is critical in classifying benign and malignant tumors in breasts.

Magnetic response imaging is screening type of which is more detailed than X-Ray and it takes across the image for the body by using a magnetic field and can be utilized with women in age thirty years old. Magnetic response imaging (MRI) was considered as a mode of screening to detect breast cancer which was appeared in the late 80's [8]. This method of testing is not recommended for women with small age and no high risk in cancer less than 15% but is essential in high-risk cases because of its high cost. The primary type of database used in this screening method is Dynamic contrast-enhanced magnetic resonance imaging (DCE-MRI) and segment image getting from this database by two ways normal segmentation, or 3D segmentation [13]. There is another kind of screening called computed tomography (CT) but it was not used in a wide range although it enhanced the image for breast tissue than other methods. The idea of detecting unhealthy cases for breast cancer and giving good results is to merge it with another type of screening such as MRI or mammogram [14,15].

Another method for screening breast cancer, called thermography, was appeared in 1982. This technique for detecting cancer in breast was called thermogram image. The mammography screening procedure is considered as the most common method for detecting cancer in the breast. It has some drawbacks in discovering cases such as that; it is not useful for younger women than forty years old. Thermogram screening procedure is a modern method for breast cancer detection and it is depending on the heat of the body. In normal cases, there is symmetrical in a thermal screen for both breasts, but in abnormal, there is no symmetrical in a thermal screen for both breasts. It is fast; economy; free of risk; less of painful besides that it has no ionizing radiation [16]. A shared database was used for this mode of thermal breast image named Mastology Research with Infrared Image-DMR-IR [17].

This study focuses on reviewing the methods and techniques for detecting of the cancer in the breast using thermography images. The aim of this study is knowing the methods applied in each stage and its results. The paper discusses stages of breast thermal image acquisition, preprocessing and segmentation, feature extraction, and classification for different methods. Finally, the current research directions are discussed.

2 Cancer Detection and Classification Methods

Detecting cancer in the breast and classifying it to normal and abnormal cases using thermography image technique is firstly converting RGB thermal image to grayscale. The state-of-the-art methods consist of five main steps of image acquisition, image preprocessing and segmentation, extract features, and finally classify image to a normal or abnormal case.

2.1 Breast Thermal Image Acquisition

The first step is thermal image acquisition for breast; there are some hints must be taken before the patient take thermogram images. Avoid smoking, drinking caffeine or alcohol, before two hours and don't apply any cream, lotion, and powder. The thermal image is taken at room temperature between 18° to 23 °C [17] with three views namely frontal, oblique and lateral [18]. There are two types of protocols may be static or dynamic [19]. It is preferred to down temperature dynamic to avoid static errors.

2.2 Preprocessing and Segmentation

The main idea for preprocessing depending on converting RGB scale mode to grayscale mode, LAB mode or HSV mode. The main idea for segmentation step is right, and left breasts separated from a thermal image which also includes the background, arms, and neck after converting to grayscale. There is more than a method used for segmentation such as region-based, threshold-based, and edge-based techniques. Methods such as [5,19–23] used Canny edge detector, Sobel edge detector, detection corner edge and other segmentation methods. Hough transform technique was used in [19,20,24,25]. Other methods are depending on manual, automatic, cropping and background removal such as [26,27]. The C-Fuzzy technique used by [8,28] while threshold techniques used by [19,29]. Morphological operators and best cluster selection were used in [18,29,30].

Block matching, and 3D filtering (BM3D) technique [15] reduces image noise features in preprocessing stage which performed by grouping and collaborative filtering. First divided the image into groups of blocking size $N \times N$ noisy blocks by using 3D transformation to make collaborative filtering in the second stage they make the separation for breast tissue than background by choosing block after removing non-breast region this block called Region of interest (ROI). The author depends on the select boundary of a breast by using Sobel edge detector to determine the edges. Authors in [19] use a computer to identify ROI then segmentation infrared image. They use a semi-automated segmentation to process image. First, they try to determine the outside boundary for a breast. They use Canny edge detector to detect the edge. The algorithm steps are to find the breast edge with adaptive thresholds by Canny edge detector, obtain closed contours by Morphological bridging operations, discard bigger contour regions and regions in the upper breasts because is not like as nipple area, and finally, choose the region which closes to nipples. They then analyze the image by using backpropagation neural network. To detect breast cancer apply preprocessing they are merging threshold with the edge based technique by converting RGB IR image to grayscale then segment breast by ROI to the right and left breast.

Depending on the fact that each region of the body has its color and each color refer to its temperature, authors in [17] use asymmetry by changing color thermos image into gray to processing image. To make segmentation, they use Hough transform method which describes the objects in the image as mathematical functions of parameters by transformation edge pixels to new coordinates by this step calculations time become less. Processing the thermal image by Fuzzy c-means (FCM) algorithm can be based on five steps of smoothing, gradient finding, non-maximum suppression, double thresholding, and edge tracking by hysteresis. A Gaussian filter is used to remove noise for smoothing image. Gradient finding used to find the edges and determine it. Non-maximum suppression used to convert blurred edges to sharp edges, and double thresholding used a threshold to separate edge than an image. Edge tracking by hysteresis used binary large object analysis. Finally, they are making cluster by FCM algorithm [8,28]. Using the filter in the spatial domain to smooth, enhance, modify, and adjust the contrast of thermal images in preprocessing is proposed in [29]. Automatic segmentation step based on normalized histogram and morphology by reconstruction and performed by threshold method was then used to segment right and left breast [31].

Preprocessing using National Television System Committee (NTSC) to convert the 24-bit thermal image into 8-bit grayscale and removing unwanted labels and tags was proposed in [26]. Depending on the distance between camera and body, a method was presented in [27]. This method consists of ROI segmentation by extract breast region from the rest of body, ROI enhancement considered as an important step in image processing. Segmentation step was used to divide images into three parts of background removal by applying Otsu's thresholding method, inframammary fold detection to convert lower part of the breast (bottom half) into binary, and axilla detection based on Canny edge detector to detect body boundary and extract. Using CAD system in [28] was consists of a thermal grayscale image in preprocessed and filter image from noise, pre-segmentation and post-segmentation using Fast Fuzzy c-means to segment Neutrosophic Sets image, and ROI. Detecting cancer in the breast using resizing and median filter to remove any noise from the image in preprocessing and the segmentation step based on best cluster selection and operator morphology was presented in [30].

2.3 Feature Extraction

Feature extraction used when there was asymmetric in heat detected for breast tissues of women body. Feature extraction types are statistical features like mean, median, standard deviation, minimum and maximum temperature values recommended when there is an asymmetry between quadrants of the breast. Histogram-based features such as mean, variance, skewness and kurtosis which extracted when right and left breasts have asymmetry and gray level co-occurrence features and gray level run-length features which used to classify normal and abnormal cases. Then after segmentation using gray-level co-occurrence matrix (GLCM) which used to calculate this texture features entropy, energy, a difference of variance and contrast [15]. In [19], some features were extracted like range temperature, mean temperature, standard deviation and the quantization of higher tone in an eight-level posterization.

Co-occurrence matrix and run-length matrix extracted texture feature which divided into two categories structural and statistical [32]. In that method, by using the curvelet domain for infrared breast image a series of statistical and texture features extracted. In feature extraction, some statistical features used like mean, median, mode, variance and standard deviation then analysis it by using gray-level GLCM or gray-level spatial dependence matrix [18]. When they note any asymmetric in the thermal image, they extracted it by skewness, temperature variation, and kurtosis which considered statistical feature extraction [20]. In [29], feature extraction making with a simple texture like spectrum, statistical, invariant moments, wavelets, Gabor, Fourier, and curvet transform. In [26], feature extraction collect information about tumor by discrete wavelet transform. Feature extraction by using the grey level GLCM which known with another name called gray-level spatial dependence matrix this considered as an important method for making extraction [33]. Some feature extraction by using the group of techniques GLCM, Gabor filter, and introductory statistics are used [28]. From breast ROI several features like statistical, texture and Gabor features are extracted [30].

2.4 Classification

The classification used to detect healthy and unhealthy cases after extraction of the features by a more different method. The most familiar method between them is support vector machine (SVM), which has some advantages such as flexibility, and implicitly. The main drawback is the accuracy of the results. This is also known as an automatic classification method which were used by

[18,19,27,28,30,32,33]. In [8], Adaboost classifier technique was used, which was easy to program, versatile and fast. K-nearest neighbor classifier was used by [24,29,33] which was helpful for large data but expensive and determination of K-value was not accurate. While the neural network was used by [22,26] and it was robust and good in large data but time was consumed. There is also nave Bayes classifier used in [33] which was easy for a small date and gave good results. It is recommended to make classification in the new trends such as a neural network and nave Bayes classifier.

3 Future Directions

This section discusses the new trends and different directions to improve the current state-of-the-art methods. Table 1 introduced the limitations and database information for some of the state-of-the-art methods. Some of these methods will be discussed in more details in this section. To give results more than 90.9% hit ratio in [34], other features extracted as well as other clustering algorithms and clustering validity indexes should be tested. Blood perfusion inside human breast is another uncertainty that needs to be carefully studied in the future [35]. Depending on increasing input parameters and using a large set data can improve the results. The accuracy of 88.10%, the sensitivity of 85.71%, and specificity of 90.48% in the presented method in [32] can be improved using an extensive database and extract better texture features.

There is a promising classifier utilized in [29] which achieved 94% accuracy using the KNN classifier compared with another classifier such as support vector machine (SVM) and neural network (NN). A method in [26] achieved an accuracy of 90.48%, sensitivity of 87.6%, and specificity of 89.73%. To enhance these results, one should solve some problems such as limited collection databases, during segmentation after remove background appears some impurities. Removes by gray level reconstruction technique but using another suitable soft computing technique will enhance accuracy.

To achieve more efficiency another segmentation method to obtain the quantitative measure aiming to bring out the abnormality present in the breast tissues [36]. Using more of a feature than used [21]. Using a large set of databases to test reliability [27,28]. Analysis database by developing computing method [17]. Using a larger dataset to check the reliability of CAD system and using modified different swarms' algorithms to enhance the system in [30]. Merging K-means clustering with global cluster doing dangerous work and some problem appears such as difficult to predict K-Value, a different final is. The result is too different initial partitions, and in the original data when merging them, it produces bad work when it has various size or density to solve these problems Fuzzy c-means and level methods for segmentation developed [24].

Table 1. Limitations and database for some of the state-of-the-art methods.

Author(s)	Method	Database	Limitations
Krawczyk et al. [4]	Multiple classifier system	146 breast thermograms (29 malignant, 117 benign)/public	Sensitivity was low (79.86%) because used features should be increased
de Oliveira et al. [5]	Automatic segmentation	180 thermal image/public	Used one side (lateral breast)
Lanisa et al. [10]	Edge detection and color morphology	50 thermal image/public	Database should be increased to improve results
Prabha et al. [16]	Block matching and 3D filtering technique	20 image (abnormal)/public	Use four feature extraction and Database should be increased to improve the results
Francis et al. [18]	Curvelet transform based feature extraction	22 thermal image/private	The method gave good results in Accuracy and specificity, but it gave low sensitivity
Kapoor et al. [20]	Asymmetry analysis of thermogram using segmentation	One volunteer/private	No database to get acceptable results
Gogoi et al. [21]	Statistical feature analysis	279 thermal image/public	More analysis is needed
Mohamed et al. [22]	Neural network (NN) classifier	206 image (187 normal, 19 abnormal)/public	Cutting breast in image was not accurate
Shahari et al. [24]	Color analysis using K-means clustering technique	Number of patient was not specified/public	Didn't work well with cluster global with various size & density
Sedong et al. [25]	Thermal infrared image analysis	250 thermal images/private	Method need to enhanced to increase the accuracy
Pramanik et al. [26]	Wavelet based thermogram analysis	306 image (123 unhealthy, 183 healthy)/public	database need to increased with enhanced method to improve the accuracy
Ali et al. [27]	An automatic segmentation methods	63 thermal image (29 healthy, 34 malignant)/public	Method reliability need to be checked
Gaber et al. [28]	Use neutrosophic sets and fuzzy c-means to detect breast cancer	63 thermal image (29 healthy, 34 malignant)/public	Method reliability need to be checked
Mejia et al. [29]	Based on texture descriptors	9 cases/public	Database was limited and more comparisons are needed
Sayed et al. [30]	Bio-inspired swarm optimization	63 thermal image (29 healthy, 34 malignant)/public	Different swarm algorithms should be tested
Acharya et al. [32]	Use texture features and support vector machine	50 image (25 normal, 25 cancerous)/public	Number of features extracted should be increased
Milosevic et al. [33]	Detection using texture features and minimum variance quantization	40 images (26 normal, 14 abnormal)/public	Results can be improved by other feature extraction and classification methods

4 Conclusion

In this paper, a review of the state-of-the-art methods and techniques used for detecting and classifying the breast cancer using thermography images was introduced. The cancer was considered as the leading cause of death among people spatially women and was basically generated from the uncontrolled growth of cells to collect them together to construct tumor. Detecting breast cancer is depending on asymmetry in temperature between breasts. The thermogram was more proper screening than other types of screening methods such as the mammogram, ultrasound, and magnetic resonance imaging. In this work, the healthy and unhealthy cases of breast cancer were classified based on the acquisition, preprocessing, segmentation, feature extraction and classification methods. As a future work, a method based on color thermal image will be to developed to detect breast cancer by identifying normal and abnormal cases.

References

1. American cancer society: Cancer facts & figures 2017. https://www.cancer.org/research/cancer-facts-statistics/all-cancer-facts-figures/cancer-facts-figures-2017.html. Accessed 3 Nov 2017
2. American cancer society: Breast cancer facts & figures 2017–2018. https://www.cancer.org/research/cancer-facts-statistics/breast-cancer-facts-figures.html. Accessed 3 Nov 2017
3. Domnguez, A.R., Nandi, A.K.: Detection of masses in mammograms via statistically based enhancement, multilevel-thresholding segmentation, and region selection. Comput. Med. Imaging Graph. **32**(4), 304–315 (2008)
4. Krawczyk, B., Schaefer, G.: Breast thermogram analysis using classifier ensembles and image symmetry features. IEEE Syst. J. **8**(3), 921–928 (2014)
5. de Oliveira, J.P.S., Conci, A., Prez, M.G., Andaluz, V.H.: Segmentation of infrared images: a new technology for early detection of breast diseases. In: 2015 IEEE International Conference on Industrial Technology (ICIT), pp. 1765–1771 (2015)
6. Qi, H., Diakides, N.A.: Thermal infrared imaging in early breast cancer detection-a survey of recent research. In: Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (IEEE Cat. No. 03CH37439), vol. 2, pp. 1109–1112 (2003)
7. Selvathi, D., Aarthy Poornila, A.: Deep learning techniques for breast cancer detection using medical image analysis. In: Biologically Rationalized Computing Techniques For Image Processing Applications, pp. 159–186. Springer International Publishing (2018)
8. Etehadtavakol, M., Ng, E.Y.K.: Breast thermography as a potential non-contact method in the early detection of cancer: a review. J. Mech. Med. Biol. **13**(02), 1330001 (2013)
9. Atlas, N.E., Aroussi, M.E., Wahbi, M.: Computer-aided breast cancer detection using mammograms: a review. In: 2014 Second World Conference on Complex Systems (WCCS), pp. 626–631 (2014)
10. Lanisa, N., Cheok, N.S., Wee, L.K.: Color morphology and segmentation of the breast thermography image. In: 2014 IEEE Conference on Biomedical Engineering and Sciences (IECBES), pp. 772–775 (2014)

11. Shan, J.: A fully automatic segmentation method for breast ultrasound images. Ph.D. thesis (2011)
12. Sehgal, C.M., Weinstein, S.P., Arger, P.H., Conant, E.F.: A review of breast ultrasound. J. Mammary Gland Biol. Neoplasia **11**(2), 113–123 (2006)
13. Xing, Y., Ou, Y., Englander, S., Schnall, M., Shen, D.: Simultaneous estimation and segmentation of t1 map for breast parenchyma measurement. In: 2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, pp. 332–335 (2007)
14. Nelson, T.R., Cervio, L.I., Boone, J.M., Lindfors, K.K.: Classification of breast computed tomography data. Med. Phys. **35**(3), 1078–1086 (2008)
15. Jalalian, A., Mashohor, S., Mahmud, R., Karasfi, B., Iqbal Saripan, M., Ramli, A.R.: Computer-assisted diagnosis system for breast cancer in computed tomography laser mammography (CTLM). J. Digit. Imaging **30**, 796–811 (2017)
16. Prabha, S., Sujatha, C.M., Ramakrishnan, S.: Asymmetry analysis of breast thermograms using bm3d technique and statistical texture features. In: 2014 International Conference on Informatics, Electronics Vision (ICIEV), pp. 1–4 (2014)
17. Silva, L.F., Saade, D.C.M., Sequeiros, G.O., Silva, A.C., Paiva, A.C., Bravo, R.S., Conci, A.: A new database for breast research with infrared image. J. Med. Imaging Health Inform. **4**(1), 92–100 (2014)
18. Francis, S.V., Sasikala, M., Saranya, S.: Detection of breast abnormality from thermograms using curvelet transform based feature extraction. J. Med. Syst. **38**(4), 23 (2014)
19. Borchartt, T.B., Conci, A., Lima, R.C., Resmini, R., Sanchez, A.: Breast thermography from an image processing viewpoint: a survey. Signal Process. **93**(10), 2785–2803 (2013)
20. Kapoor, P., Prasad, S.V.A.V.: Image processing for early diagnosis of breast cancer using infrared images. In: 2010 The 2nd International Conference on Computer and Automation Engineering (ICCAE), vol. 3, pp. 564–566 (2010)
21. Gogoi, U.R., Majumdar, G., Bhowmik, M.K., Ghosh, A.K., Bhattacharjee, D.: Breast abnormality detection through statistical feature analysis using infrared thermograms. In: 2015 International Symposium on Advanced Computing and Communication (ISACC), pp. 258–265 (2015)
22. Mohamed, N.A.E.R.: Breast cancer risk detection using digital infrared thermal images. Int. J. Bioinform. Biomed. Eng. **1**(2), 185–194 (2015)
23. Ibrahim, A., Gaber, T., Horiuchi, T., Snasel, V., Hassanien, A.E.: Human thermal face extraction based on superpixel technique. In: Proceedings of the 1st International Conference on Advanced Intelligent System and Informatics (AISI 2015), pp. 163–172. Springer International Publishing (2016)
24. Shahari, S., Wakankar, A.: Color analysis of thermograms for breast cancer detection. In: 2015 International Conference on Industrial Instrumentation and Control (ICIC), pp. 1577–1581 (2015)
25. Sedong, M., Jiyoung, H., Youngsun, K., Yunyoung, N., Preap, L., Bong-Keun, J., Dongik, O., Wonhan, S.: Thermal infrared image analysis for breast cancer detection. KSII Trans. Internet Inf. Syst. **11**(2), 1134–1147 (2017)
26. Pramanik, S., Bhattacharjee, D., Nasipuri, M.: Wavelet based thermogram analysis for breast cancer detection. In: 2015 International Symposium on Advanced Computing and Communication (ISACC), pp. 205–212 (2015)
27. Ali, M.A.S., Sayed, G.I., Gaber, T., Hassanien, A.E., Snasel, V., Silva, L.F.: Detection of breast abnormalities of thermograms based on a new segmentation method. In: 2015 Federated Conference on Computer Science and Information Systems (FedCSIS), pp. 255–261 (2015)

28. Gaber, T., Ismail, G., Anter, A., Soliman, M., Ali, M., Semary, N., Hassanien, A.E., Snasel, V.: Thermogram breast cancer prediction approach based on neutrosophic sets and fuzzy c-means algorithm. In: 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 4254–4257 (2015)
29. Mejia, T.M., Prez, M.G., Andaluz, V.H., Conci, A.: Automatic segmentation and analysis of thermograms using texture descriptors for breast cancer detection. In: 2015 Asia-Pacific Conference on Computer Aided System Engineering, pp. 24–29 (2015)
30. Sayed, G.I., Soliman, M., Hassanien, A.E.: Bio-inspired swarm techniques for thermogram breast cancer detection. In: Medical Imaging in Clinical Applications: Algorithmic and Computer-Based Approaches, pp. 487–506. Springer International Publishing (2016)
31. Garduno-Ramon, M.A., Vega-Mancilla, S.G., Morales-Henandez, L.A., Osornio-Rios, R.A.: Supportive noninvasive tool for the diagnosis of breast cancer using a thermographic camera as sensor. Sensors **17**(3), E497 (2017)
32. Acharya, U.R., Ng, E.Y.K., Tan, J.H., Sree, S.V.: Thermography based breast cancer detection using texture features and support vector machine. J. Med. Syst. **36**(3), 1503–1510 (2012)
33. Milosevic, M., Jankovic, D., Peulic, A.: Thermography based breast cancer detection using texture features and minimum variance quantization. EXCLI J. **13**, 1204–1215 (2014)
34. Silva, L.F., Sequeiros, G.O., Santos, M.L.O., Fontes, C.A.P., Muchaluat-Saade, D.C., Conci, A.: Thermal signal analysis for breast cancer risk verification. Stud. Health Technol. Inform. **216**, 746–750 (2015)
35. Li, Y., Fahimi, B.: Thermal analysis of multiple-antenna-excited breast model for breast cancer detection. In: 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 1058–1061 (2016)
36. Suganthi, S., Ramakrishnan, S.: Anisotropic diffusion filter based edge enhancement for segmentation of breast thermogram using level sets. Biomed. Signal Process. Control **10**(Supplement C), 128–136 (2014)

An Optimized K-Nearest Neighbor Algorithm for Extending Wireless Sensor Network Lifetime

Mohammed M. Ahmed[1,3](✉), Ayman Taha[2], Aboul Ella Hassanien[2,3], and Ehab Hassanien[2]

[1] Faculty of Computers and Information, Minia University, Minya, Egypt
mohmed.mostafa111@yahoo.com
[2] Faculty of Computers and Information, Cairo University, Giza, Egypt
[3] Scientific Research Group in Egypt (SRGE), Giza, Egypt
http://www.egyptscience.net

Abstract. This paper presents an optimized K-nearest neighbors (KNNs) classification algorithm using the metaheuristic whale optimization to searches for sink node in wireless sensor networks. Sink node aggregate data from all sensor nodes and reducing the energy consumption network to prolong network lifetime. To reach aforementioned, a fitness function has formulated to choose the best location of sink node with high residual neighbor's sensor nodes energy to leads to maximizing the network lifetime. Eventually, the experimental results have been conducted whereas sensor nodes are propagated in a random location within the desired network area. The system has 11% improvement on the network's energy consumption that increases the lifetime of the network.

Keywords: K-nearest neighbor algorithm · Energy-efficient Classification · Wireless sensor networks · Swarm optimization

1 Introduction

In recent advances, the development of Wireless Sensor Networks (WSNs) [1] is expanding rapidly. It consists of huge number of sensor nodes that are with low cost, small volume, wireless communication, data processing ability, sensing, storage, and energy resources. There are various applications of WSNs, such as home security, surveillance, disaster relief, healthcare, and environmental monitoring [2]. WSN contains base-station with one or more sinks that aggregate data from all sensor nodes and send to sink node. Sink node location can actively decreases the energy consumption and increase the lifetime of the network by reducing the distance between the sensor and sink node [3,4]. A sink node was designated device similar to the regular sensor nodes but with more power. The main task of the sink node in wireless sensor networks (WSNs) is forwarding the messages directly to both of the sink nodes and the farthest one to save their energy. Many sensor nodes will become quickly unable to communicate with the base station, and the network becomes nonoperational. Consequently, the choice of the best location of sink node to receive all messages from sensor nodes without consuming their energies suddenly is a big challenge in WSNs.

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 506–515, 2018.
https://doi.org/10.1007/978-3-319-74690-6_50

Network energy and lifetime introduced by Chen and Li superior the energy-oriented strategy presented by Hou et al. [5] regarding networks lifetime. A single objective swarm such as a discrete version of the whale optimization algorithm was presented in [6] to determine the active nodes which cover all nodes and inactive nodes in network topology to prolong the WSNs lifetime. Also, in [7] introduced an algorithm to solve the multiple base station locations. To achieve the balance among clusters over which existing sink location for small to medium scale WSNs, Slama et al. have utilized a graph partitioning techniques [8], in [9] proposed a hybrid algorithm of both Simulated Annealing and Bee Algorithm for a weighted minimal spanning tree (BASA-WMST) and another bio-inspired techniques to build that topology such as the PSO-minimum spanning tree-based topology control scheme [10], which called the non-dominated discrete particle swarm optimization (NDPSO) also in [11] proposed method to determine position of sink node with reduce the number of active nodes to prolong network's lifetime via PSO. Most of simulations of WSN use the sink node that is placed at the center of the region. Most studies and simulations were executed based on this placement strategy. Efrat et al. [12] and Luo and Hubaux [13] have proposed model that called the P-Median Problem (PMP) to determine the sink node placement. Also, in [13] has proved that the center of the circle is the optimal position for a base station in WSNs, but the conclusion is only suitable for the uniform deployment of nodes. In [14] is proposed method to choose position of sink node to maximize the weight of data flows to reduce the energy consumption.

This work proposes whale optimization algorithm based KNN to better solve some the problem of finding the best location of single sink node in with reducing energy consumption to prolong network lifetime in WSNs environment. Moreover, after choosing the location of sink node in the network determines best of nearest neighbors nodes. Finally, the performance of the proposed WOA-KNN is compared with another algorithm such as particle swarm optimization based k-nearest neighbor (PSO-KNN).

The structure of the paper is organized as follows: we present the network model and assumptions and methods that used In Sect. 2, overview proposed algorithm, in Sect. 2.3 the proposed Whale optimization based topology control algorithm, In Sect. 4, simulation results are considered, and we conclude the paper in Sect. 5.

2 Preliminaries

2.1 Network Model and Assumptions

Assume wireless sensor network with a set of sensor nodes that consists of an active and inactive node and a single sink node that gather data from all sensor nodes in a network. Sensor nodes are randomly distributed in a given area $R = L \times L$, where L is the side length. The sensor nodes have a power source, bandwidth and memory, and they might sensor down from the network at any time due to limited battery lifetime. We assume that the sink node (base station)

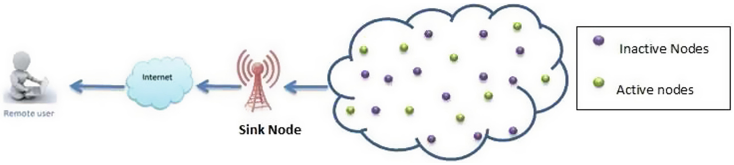


Fig. 1. Network Model architecture of wireless sensor network.

is far away from the nodes, but it is connected with a set of sensor nodes. In Fig. 1 illustrate Network Model and assumptions that used in this work.

Assumptions. We consider the following assumptions about the sensor network model N:

- Let S be a set of sensor nodes that are distributed randomly and uniformly in a given area R. All sensor nodes have the same capabilities such as mobility, homogeneous, limited memory and power.
- Let $AN = AN, ..., AN_n$ be the set of active nodes. All active nodes have the amount of memory, power, and bandwidth.
- Let Sk be the sink node that collects data from all sensor nodes.

2.2 Whale Optimization Algorithm

Mirjalili et al. proposed Whale Optimization Algorithm (WOA) [15], is considered one of swam intelligent application [16] that is a novel nature-inspired meta-heuristic optimization algorithm, whales swim around prey within a shrinking circle and along a spiral-shaped path simultaneously in order to create bubbles along a circle or '9' -shaped path. To simulate this behavior in WOA, their formulations are designed as follows:

Shrinking Encircling Preys: The target prey and the other search agents try to update their positions towards it. This behavior is represented by the following formula:

$$\overrightarrow{X}(t+1) = \overrightarrow{X}(t) - A.\overrightarrow{D} \tag{1}$$

$$\overrightarrow{D} = |C\overrightarrow{X}^*(t) - \overrightarrow{X}(t)| \tag{2}$$

$$A = 2.a.r - a \tag{3}$$

$$C = 2.r \tag{4}$$

Where$\overrightarrow{X}$ is the historically best position, $\overrightarrow{X}$ is a whale position and t indicates the current iteration. a is linearly reduced from 2 to 0 and r is a random in the range of [0,1]. The sign || denotes the absolute value.

Spiral Bubble-Net Feeding Maneuver: A spiral equation is used between the position of whale and prey as follows:

$$\overrightarrow{X}(t+1) = e^{bk}.cos(2\pi k).D' - \overrightarrow{X}^*(t) \tag{5}$$

$$D' = |\overrightarrow{X^*}(t) - \overrightarrow{X}(t)| \tag{6}$$

Where b is a constant, and k is a random in the range of $[-1, 1]$.

Search for Prey: The search agent is updated according to a randomly chosen search agent instead of the best search agent:

$$\overrightarrow{D}^{"} = |C.\overrightarrow{X(t)}_{rand} - \overrightarrow{X}(t)| \tag{7}$$

$$\overrightarrow{X}(t+1) = \overrightarrow{X(t)}_{rand} - A.\overrightarrow{D}^{"}| \tag{8}$$

Where $\overrightarrow{X(t)}_{rand}$ is selected randomly from whales in the current iteration. Finally, follows these conditions:

- $|A| > 1$ enforces exploration to WOA algorithm to find out global optimum avoids local optima.
- $|A| < 1$ For updating the position of best current search agent selected.

2.3 K-Nearest Neighbor (K-NN)

The K-nearest neighbor (K-NN) method proposed by Fix and Hodges [17] that was considered of the nonparametric methods used for classification of new object based on training samples and attributes. The K-NN is considered a supervised learning algorithm that a new instance query result is classified based on the K-nearest neighbor category [18]. The K-NN method applied in many areas: artificial intelligence, pattern recognition, statistical estimation, feature selection and categorical problems. The advantage of KNN is east to implement and simple. K-NN is not negatively affected when training data are large, and indifferent to noisy training data. The need to determine parameter K is regard disadvantage of the K-NN method, calculate the distances between the query instance and all the training samples, sort the distances and determine the nearest neighbors based on the Kth minimum distance, additionally determine the categories of the nearest neighbors.

The KNN search problem compose of searching the k nearest neighbors of each point in the reference. Commonly, the Euclidean or the Manhattan distance is used but any other distance can be used instead such as the Chebyshev norm or the Mahalanobis distance. The brute force algorithm (BF) also called exhaustive search is considered method to search the KNN. the BF algorithm is the following:

- 1. Compute all the distances.
- 2. Sort the computed distances.
- 3. Select the k reference points corresponding to the k smallest distances.
- 4. Repeat steps 1. to 3.

The aim from this technique KNN algorithm is to optimized by whale optimization algorithm in order to determine best location of sink node with best high residual neighbor's sensor nodes.

3 The Proposed WOA-KNN Algorithm

Improving the accuracy of k-NN classifier algorithm is the target of this study. To improve the k-NN classification the best solution is that best of k-nearest neighbors for sink node that selected through WOA. For this propose, first contribution of the paper has focused on the WOA. WOA is one of the best and popular search tools.

Firstly, the data set of network has been split to the active nodes and inactive nodes. Next, maximum of iteration has been determined to validate the k-NN so. Then in each iteration, the WOA procedure is called. In WOA algorithm, a population of N Whales has been produced randomly and the fitness function of population (Whales) is computed. Note that each whale introduces a real values in range [0,1] for each dimension. After that, fitness value of each whale is calculated. Then evaluated whales are used in evolutionary progress. The cycle of WOA approach described in previous Sect. 2.2. The evolutionary process has continued until the conditions are satisfied. For this propose, Whales returns a vector with the best real values in range [0,1]. After that, the weighted are stored

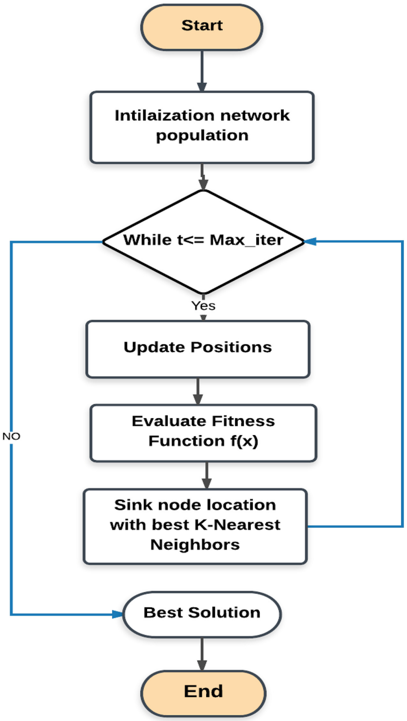

Start
Intilaization network population
While t<= Max_iter
Yes
NO
Update Positions
Evaluate Fitness Function f(x)
Sink node location with best K-Nearest Neighbors
Best Solution
End

Fig. 2. The flowchart of Whale Optimization Algorithm based K-Nearest Neighbor (WOA-KNN).

to the k-NN algorithm for classification. Note that all weights will be computed the distances. After 100 iterations, best solution are given and have been used in the k-NN.

This section describes the design and the implementation of Whale optimization algorithm based K-nearest neighbors algorithm for wireless sensor networks as shown in Fig. 2. During implementation of WOA according to fitness function that clarify in Eq. 9 after choose best of k parameter that clarify number of nearest neighbors of sink node position with high residual energy in order to Maintain on power of network.

$$f(w_i) = \alpha_1 d_w + \alpha_2 \sum_{i=1}^{N_w} E_{w_i} \tag{9}$$

Where N_w is the set of neighbors of a node w, E_{w_i} refers to the the residual energy within a neighbor node w and d_w is the Euclidean distance between the position of the node w and the center of network. One drawback of Eq. 9 is the fairness between nodes; since the nodes with low energy with a high number of nodes that covered from it.

4 Experimental Results

The proposed approach has been evaluated using ten datasets of WSN that was implemented and evaluated using a Java-based simulation tool called Atarraya [19]. The proposed WOA-KNN model was implemented using MATLAB. In Table 1 obtained simulation parameters that were adjusted for the experimental scenarios.

The simulations of nodes are assumed to mimic the characteristics of simple sensors with the energy model that defined in [20]. The performance analysis

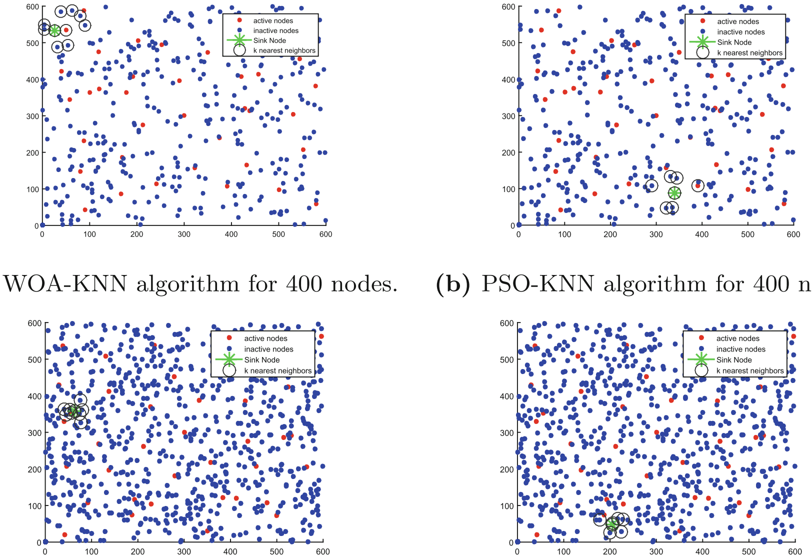
Table 1. Atarraya simulation parameter.

Parameter	Value
Deployment area	600 m * 600 m
Number of nodes	200, 400, 600..., 2000
Sensor node model	Simple
Node communication	Range 100 m
Node sensing	Range 20 m
Node location distribution	Uniform
Node energy distribution	Uniform
Max energy	1000 milliamperes-hour(mA-h)
α_1	0.5
α_2	0.3

calculated the mean of average different runs of the algorithm that is calculated for both algorithms the WOA-KNN and PSO-KNN, Table 2 summarizes all obtained results, where N, AN, K, L, EC and T means the network size, number of active nodes, number of nearest neighbors for sink node that selected, Lifetime network, Energy consumption and total network energy respectively.

Table 2. Results obtained from the WOA-KNN and PSO-KNN algorithm.

N	AN	T	WOA-K	PSO-K	WOA-L	PSO-L	WOA-EC	PSO-EC
200	39	95823	8	7	1453	1354	10546	12548
400	38	98357	9	6	1632	1573	13684	14625
600	40	106354	7	8	1475	1374	12568	12934
800	48	127862	9	7	1524	1354	13476	14022
1000	42	131425	8	10	1624	1397	16245	16834
1200	46	157332	12	11	1538	1426	18456	20045
1400	45	149652	11	13	1504	1398	17896	19357
1600	49	173589	12	11	1425	1243	16972	20279
1800	47	163471	14	12	1684	1425	19564	22687
2000	48	182463	13	11	1614	1375	18674	23745



(a) WOA-KNN algorithm for 400 nodes. **(b)** PSO-KNN algorithm for 400 nodes.

(c) WOA-KNN algorithm for 800 nodes. **(d)** PSO-KNN algorithm for 800 nodes.

Fig. 3. Sink node location with best nearest neighbors of the WOA-KNN and PSO-KNN for 400 and 800 nodes.

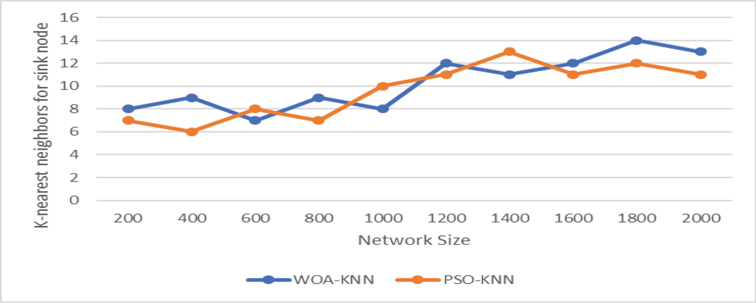


Fig. 4. No. of K-nearest neighbors.

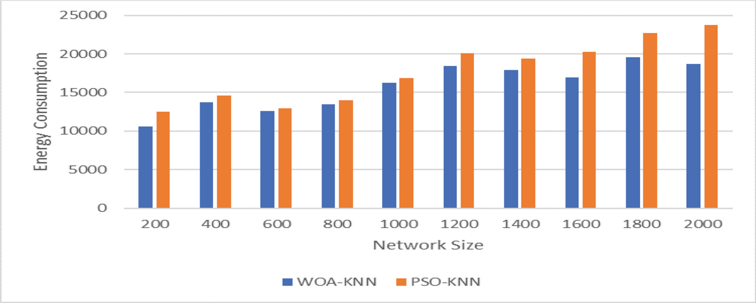


Fig. 5. Energy consumption.

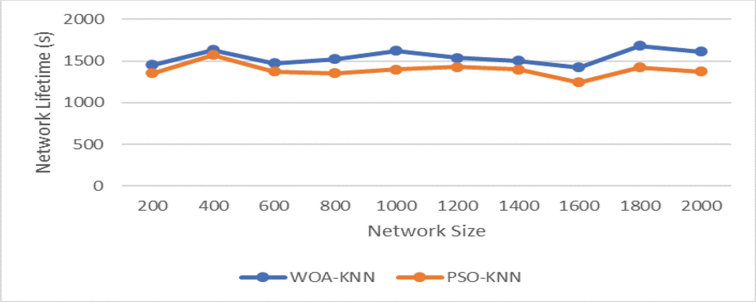


Fig. 6. Lifetime network.

Results of the WOA-KNN algorithm are shown in Fig. 3 for network size 400 and 800 nodes, active nodes and inactive nodes clarify by a red color and a blue color in graph respectively and green color represents sink node for network. The main target of constructing a reduced network energy has been achieved where determine best location of sink node using the proposed algorithm WOA-KNN compared to the PSO-KNN algorithm. Figure 4 shows the number of

K nearest neighbors for WOA-KNN and PSO-KNN. Figure 5 illustrates energy consumption for a network and Fig. 6 illustrates the lifetime between both algorithms.

5 Conclusion

This work proposed method to improve K-nearest Neighbor via whale optimization algorithm that used it in Wireless Sensor Networks (WSNs) sink node location problem. To solve this problem of finding the best location of single sink node with optimal number of k parameter that determines best of neighbor's sink node that has high residual energy in order to reducing energy consumption to extending network lifetime in WSNs environment. We proposed WOA-KNN to choose optimal location of single sink node with neighbor's high residual energy that gathers data from all active nodes in network, After getting the location of the sink node using the proposed WOA-KNN reconstructs network according to sink node position. The proposed algorithm save the energy consumption approximately by 11% compared with the well-known algorithm PSO-KNN.

References

1. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. Comput. Netw. **38**(4), 393–422 (2002)
2. Estrin, D., Govindan, R., Heidemann, J., Kumar, S.: Next century challenges: scalable coordination in sensor networks. In: Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking, pp. 263–270. ACM (1999)
3. Vass, D., Vincze, Z., Vida, R., Vidács, A.: Energy efficiency in wireless sensor networks using mobile base station. In: EUNICE 2005: Networks and Applications Towards a Ubiquitously Connected World, pp. 173–186. Springer (2006)
4. Kim, H., Seok, Y., Choi, N., Choi, Y., Kwon, T.: Optimal multi-sink positioning and energy-efficient routing in wireless sensor networks. In: International Conference on Information Networking, pp. 264–274. Springer (2005)
5. Hou, Y.T., Shi, Y., Sherali, H.D., Midkiff, S.F.: On energy provisioning and relay node placement for wireless sensor networks. IEEE Trans. Wirel. Commun. **4**(5), 2579–2590 (2005)
6. Ahmed, M.M., Houssein, E.H., Hassanien, A.E., Taha, A., Hassanien, E.: Maximizing lifetime of wireless sensor networks based on whale optimization algorithm. In: International Conference on Advanced Intelligent Systems and Informatics, pp. 724–733. Springer (2017)
7. Vincze, Z., Fodor, K., Vida, R., Vidács, A.: Electrostatic modelling of multiple mobile sinks in wireless sensor networks. In: Proceedings of the IFIP Networking Workshop on Performance Control in Wireless Sensor Networks (PWSN 2006), Coimbra, Portugal, pp. 30–37 (2006)
8. Slama, I., Jouaber, B., Zeghlache, D.: Multiple mobile sinks deployment for energy efficiency in large scale wireless sensor networks. In: International Conference on E-Business and Telecommunications, pp. 412–427. Springer (2008)

9. Saravanan, M., Madheswaran, M.: A hybrid optimized weighted minimum spanning tree for the shortest intrapath selection in wireless sensor network. Math. Probl. Eng. **2014**, 8 (2014)
10. Mostafaei, H., Meybodi, M.R.: Maximizing lifetime of target coverage in wireless sensor networks using learning automata. Wirel. Pers. Commun. **71**(2), 1461–1477 (2013)
11. Fouad, M.M., Snasel, V., Hassanien, A.E.: Energy-aware sink node localization algorithm for wireless sensor networks. Int. J. Distrib. Sens. Netw. **11**(7), 810356 (2015)
12. Efrat, A., Har-Peled, S., Mitchell, J.S.: Approximation algorithms for two optimal location problems in sensor networks. In: 2nd International Conference on Broadband Networks, BroadNets 2005, pp. 714–723. IEEE (2005)
13. Luo, J., Hubaux, J.-P.: Joint mobility and routing for lifetime elongation in wireless sensor networks. In: INFOCOM 2005, 24th Annual Joint Conference of the IEEE Computer and Communications Societies, Proceedings IEEE, vol. 3, pp. 1735–1746. IEEE (2005)
14. Bogdanov, A., Maneva, E., Riesenfeld, S.: Power-aware base station positioning for sensor networks. In: INFOCOM 2004, Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 1. IEEE (2004)
15. Mirjalili, S., Lewis, A.: The whale optimization algorithm. Adv. Eng. Softw. **95**, 51–67 (2016)
16. Hassanien, A.E., Emary, E.: Swarm Intelligence: Principles, Advances, and Applications. CRC Press, New York (2016)
17. Fix, E., Hodges Jr., J.L.: Discriminatory analysis-nonparametric discrimination: consistency properties. California Univ Berkeley, Technical report (1951)
18. Castillo, O., Xu, L., Ao, S.-I.: Trends in Intelligent Systems and Computer Engineering. Springer, New York (2008)
19. Labrador, M.A., Wightman, P.M.: Topology Control in Wireless Sensor Networks: With a Companion Simulation Tool for Teaching and Research. Springer Science & Business Media, New York (2009)
20. Cai, Y., Li, M., Shu, W., Wu, M.-Y.: Acos: An area-based collaborative sleeping protocol for wireless sensor networks. Ad Hoc Sens. Wirel. Netw. **3**(1), 77–97 (2007)

Efficient Twitter Sentiment Analysis System with Feature Selection and Classifier Ensemble

Mohammed M. Fouad[1], Tarek F. Gharib[1](✉), and Abdulfattah S. Mashat[2]

[1] Faculty of Computer and Information Sciences, Ain Shams University, Cairo, Egypt
tfgharib@cis.asu.edu.eg

[2] Faculty of Computing and Information Technology, King Abdulaziz University,
Jeddah, Saudi Arabia

Abstract. Sentiment analysis from Twitter is one of the interesting research fields recently. It combines natural language processing techniques with the data mining approaches for building such systems. In this paper, we introduced an efficient system for Twitter sentiment analysis. The proposed system built a machine learning model for detecting positive and negative tweets. This model used different techniques to represent the input labeled tweets in the training phase using different features sets. In the classification phase, the classifier ensemble is presented with different base classifiers for more accurate results. The proposed system can be used for measuring users' opinion from their tweets which is very useful in many applications such as marketing, political polarity detection and reviewing products.

Keywords: Opinion mining · Sentiment analysis · Classifier ensemble
Feature selection · Information gain

1 Introduction

Nowadays, chatting, social media communications, blogging and micro-blogging, are the most utilized online activities on the Internet. Twitter is one of the most popular microblogging services and could be considered one of the largest user-generated content with very huge amount of structured and unstructured data. The posted tweets can express users' opinions about different topics and express their polarity towards these topics depending on the users' interests.

Sentiment analysis (also called opinion mining) uses a combination of data mining, text and web mining techniques in order to detect, extract and recognize the opinions, emotions and attitudes towards certain topics. It could be applied to different data sources such as review, blogs and news [1]. Detecting users' opinions is very useful information in many application domains such as evaluating marketing campaigns [2], reviewing movies [3], customer satisfaction [4] and many more.

Sentiment analysis in Twitter is different from other social media platforms due to many aspects: (i) users tend to use very short tweets to express their mood and status; (ii) users may use some abbreviations, emoticons to save up some characters; (iii) many linguistic representation challenges arise from feature engineering issues [5]. In this

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 516–527, 2018.
https://doi.org/10.1007/978-3-319-74690-6_51

paper, we studied different combinations of features sets that could be used in tweets representation efficiently. As seen in most of the text mining systems, the extracted features may cause a complex computation problem due to the huge dimension of the generated features vector. To deal with such problem, features selection techniques such as information gain and mutual information [6], etc., or features transformation methods such as feature hashing [7] and principle component analysis [8] could be applied too.

Depending on the tweet message itself, it could be considered as a positive or a negative tweet if this message contains a sentiment with its text body, otherwise it is considered a neutral one. This led us to consider the sentiment analysis system as a classification problem. In such problem, we should analyze the input tweets collections and classify them with respect to the existing sentiments in each one. Moreover, the combination of multiple classifiers (called ensemble) is used to generate a single classifier to benefit from the properties of the individual classifiers. In this paper, we applied a majority voting ensemble that combines the decision from three base classifiers.

The main objective of this paper is to propose an efficient system for Twitter sentiment analysis using the information gain as a feature selection technique and the majority voting ensemble classifier. The proposed system is implemented and its accuracy is evaluated in order to answer the following **research questions**: (i) What is the prominent feature set that achieves the highest accuracy? And (ii) Does using information gain lead to better performance or not? And (iii) Does the ensemble model provide higher accuracy than the individual classifiers? Moreover, what are the factors that affect its performance?

The remainder of this paper is organized as follows: Sect. 2 contains the most relevant work in sentiment analysis problem. Section 3 represents in brief details the proposed system for twitter sentiment analysis. Experimental results and discussion are presented in Sect. 4. Conclusions are finally drawn in Sect. 5.

2 Related Work

Sentiment analysis is proposed to discover the users' polarity towards certain subject from their comments, reviews or opinions. This topic has been applied on news articles, blogs, product reviews, micro-blogs and forums. Due to the extensive research in this area, Ravi and Ravi [9] presented a detailed survey on the tasks, the approaches and the applications of the opinion mining that included a separate section for sentiment analysis in general. Another survey provided by Kharde and Sonawane [10] that covered the techniques of the sentiment analysis on Twitter data with comparative analysis of the existing approaches.

Ghiassi et al. [11] developed a Twitter-specific lexicon for sentiment analysis by utilizing a supervised feature selection technique using n-grams and statistical analysis. Their proposed model was tested using 3440 manually collected and annotated tweets from Justin Bieber Twitter account. Their experimental results show that their proposed model slightly outperformed the standard SVM classifier and achieved 95.1% accuracy.

Selecting the prominent features set for sentiment analysis is one of the challenges with the existing sentiment analysis methods. There are several types of features that could be extracted from the tweets text, but what are the combination that achieves the highest accuracy rate. Recently, Agrawal and Mittal [12] explored various feature extraction and selection techniques to discover the prominent features in a machine learning based sentiment analysis. They combined the lexicon-based approaches with the corpus-based approaches to find the semantic orientation of all the extracted features to measure the overall polarity of the input text.

Agrawal et al. [13] studied the feature-engineering problem on twitter sentiment classification. Their feature sets combined different features such as unigram, POS-features, senti-features and tree kernel model. For the classification task, SVM was used with the different feature sets combinations. They applied their proposed system to a collection of 11,875 tweets that were manually annotated. According to their results, the feature set containing the unigram and senti-features achieved the highest accuracy rate with about 75.39%.

Most of the presented machine learning based sentiment analysis methods used a single classifier to perform the classification task. For example, Zhang et al. [14] and Mohammad et al. [15] used Support Vector Machines (SVM) algorithm, while others like Saif et al. [16] utilized the Naïve Bayes (NB) algorithm because of their good performance in text classification problems. On the other hand, classifier ensemble approach is introduced to train multiple classifiers and combine their decisions to solve the same classification problem. This approach tried to cover some of the problems of the individual classifiers by combining different classifiers tending to produce a generalized decision boundary for the classification input [17]. It is not guaranteed that the performance of the classifier ensemble is always better than the individual classifiers combined in it, but in some cases, it reduces the risk of selecting inefficient classifier with the unseen data [18].

The proposed classifier ensembles are different in the base classifiers used in each ensemble and the way it combines their decisions. For example, both Lin and Koltz [19] and Rodríguez-Penago et al. [20] used the majority voting ensemble in their work. Clark et al. [21] used the weighted voting ensemble with trained Naïve Bayes classifiers. In addition, Hassan et al. [5] proposed a bootstrap model that combined different dataset, feature and classifier parameters with utilization of about 6 base classifiers. Recently, Da Silva et al. [22] presented another combination rule. They calculated the average of the probabilities that were produced by four classifiers for each class as the final decision of the ensemble classifier.

3 The Proposed System

In this section, we will describe in brief details the components of the proposed system for twitter sentiment analysis. As shown in Fig. 1, the proposed system is running in two phases: Training and Classification phases. The purpose of the training phase is to build the classification model in order to distinguish between positive and negative tweets based on the input labeled tweets collections. In the classification phase, the trained classification model will assign positive or negative label to the new unlabeled tweets. The system contains four steps: Preprocessing, Feature Extraction, Feature Selection and the Classification Model for Sentiment Analysis.

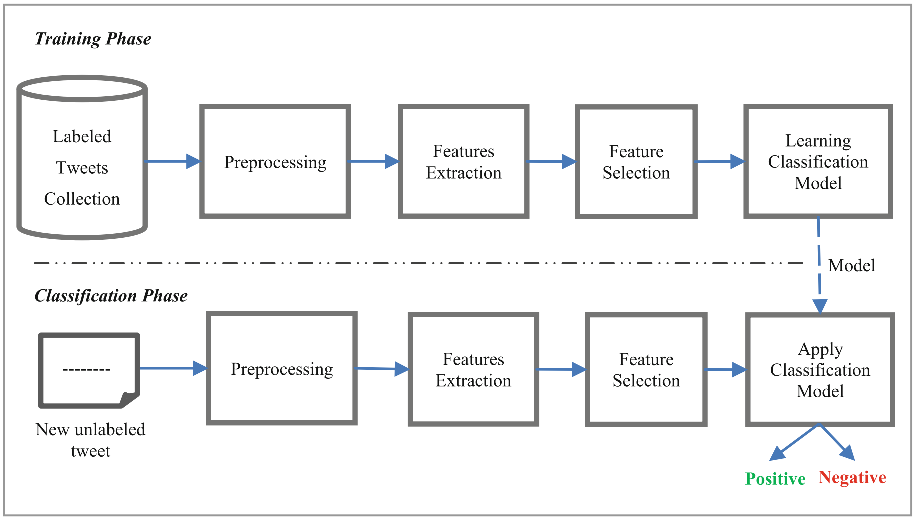


Fig. 1. Overview of the proposed system

3.1 Preprocessing

The main objective of this step is to use the natural language processing techniques to process the input tweet text and make it suitable for the next step to extract the features correctly. The detailed block diagram with example tweet for the preprocessing step is shown in Fig. 2.

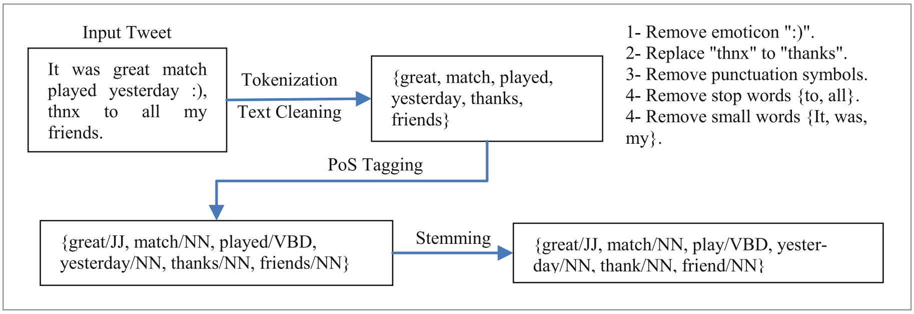

Input Tweet
It was great match played yesterday :), thnx to all my friends.
Tokenization
Text Cleaning
{great, match, played, yesterday, thanks, friends}
1- Remove emoticon ":)".
2- Replace "thnx" to "thanks".
3- Remove punctuation symbols.
4- Remove stop words {to, all}.
4- Remove small words {It, was, my}.
PoS Tagging
{great/JJ, match/NN, played/VBD, yesterday/NN, thanks/NN, friends/NN}
Stemming
{great/JJ, match/NN, play/VBD, yester-day/NN, thank/NN, friend/NN}

Fig. 2. Example for tweet text preprocessing

The preprocessing step includes four sub-steps: Tokenization, Text Cleaning, PoS Tagging and Stemming. The preprocessing started with tokenizing (splitting) the input text into separate terms (called tokens). Each token can represent word, abbreviation, hyperlink, emoticon or other punctuation symbols that could be found commonly in tweets.

The second step is the Text Cleaning step, which is responsible for removing any irrelevant textual data from the tweet content itself. As shown in Fig. 2, the example input tweet "It was great match played yesterday :), thnx to all my friends" is transformed into a list of words which is {great, match, played, yesterday, thanks, friends} after applying the tokenization and text cleaning steps.

The third step is PoS Tagging in which we extract the part of speech tags for the input text. For example, word such as "great" is tagged with "JJ" because it is an adjective word. The final step is to stem the words to their original root in order to reduce the initial set of words representing the input tweet text. For example, the word "played" is transformed into the stem word "play". The final words of the preprocessing step for the example tweet is {great/JJ, match/NN, play/VBD, yesterday/NN, thank/NN, friend/NN}.

3.2 Feature Extraction

There are several features to be extracted to represent the input tweet text. In this section, we will present the different types of features that will be used in the proposed system.

The most simple and traditional technique in text representation is to extract all the possible stemmed words, may also called terms or tokens, from the input text which is called **Bag-of-words (BoW)**. In this paper, the BoW contains all the distinct unigrams (single word terms) and bigrams (two consecutive words terms). For example tweet "great match play yesterday thank friend", the unigram features are "great", "match", "play", "yesterday", "thank" and "friend". The bigram features are "great_match", "match_play", "play_yesterday", "yesterday_thank" and "thank_friend".

Some words can express the opinion state of the writer. Words like *great*, *good*, *wonderful* and *excellent* can express positive opinion, while words such as *poor*, *bad* and *dangerous* are examples of negative opinion. In this paper, we used the opinion lexicon collected by Liu et al. [23] that contains a list of 2006 positive words and 4783 negative words. For each tweet, the positive and negative words are counted as the **Lexicon-based features**. For the example tweet, it has *two* positive words, "*great*" and "*thank*", and *zero* negative words as lexicon-based features.

During the preprocessing step, the part-of-speech tags for the extracted words are stored. We count the numbers of nouns, verbs, adjectives and adverbs as the **PoS features**. For the example tweet, the extracted PoS features are four nouns ("match", "yesterday", "thank" and "friend"), one verb ("play"), one adjective ("great") and zero adverbs.

Emoticons are some symbols that represent certain state to the writer opinion. In this step, we collected a list of commonly used emoticons used in the social media and especially in tweets. The list contains 112 positive, 77 negative and 16 neutral emoticon symbols. For each tweet in the collection, the number of the found emoticons in each state is recorded as the **Emoticons features**. For the example tweet, it has only *one* positive emoticon, which is :), and *zero* negative and neutral ones.

3.3 Feature Selection

As discussed earlier, each tweet is represented by a vector of numbers based on the extracted features. The biggest portion of these features follows to the BoW unigrams and bigrams. The dimension of this vector increased dramatically by the number of distinct terms in the input tweets collection. The curse of high dimensionality exists in most of the text processing systems including sentiment analysis ones. For this case, we used Information Gain (IG) as feature selection technique to reduce the dimension of the output feature vector. In the proposed system, the information gain weight is calculated for each feature using Eq. (1) and the features that have higher weight than 0.01 are selected.

Consider the input tweets collection with class attribute C that has two classes $\{C_1 = \text{positive and } C_2 = \text{negative}\}$. For any given feature x, the information gain (IG) is calculated by:

$$\begin{aligned} IG(x) = & -\textstyle\sum_{j=1}^{2} P\left(C_j\right) \log\left(P\left(C_j\right)\right) + P(x)\textstyle\sum_{j=1}^{2} P\left(C_j|x\right)\log\left(P\left(C_j|x\right)\right) \\ & + P(\bar{x})\textstyle\sum_{j=1}^{2} P\left(C_j|\bar{x}\right)\log\left(P\left(C_j|\bar{x}\right)\right) \end{aligned} \tag{1}$$

Where, $P\left(C_j\right)$ is the fraction of tweets labeled with class C_j, $P(x)$ is the fraction of tweets in which feature x occurs and $P\left(C_j|x\right)$ is the fraction of tweets with class C_j that has feature x.

3.4 Classification Model for Sentiment Analysis

The main step in the proposed system is to build a classification model that is able to differentiate efficiently between positive and negative labeled tweets in the training phase.

There are several machine learning algorithms that could be used in building such a model. In the proposed system, we implemented a majority voting ensemble classifier with SVM, NB and LR as base learners. These algorithms are commonly used and have great success in the text classification problems. An overview of the majority voting classifier ensemble used in the proposed system is shown in Fig. 3.

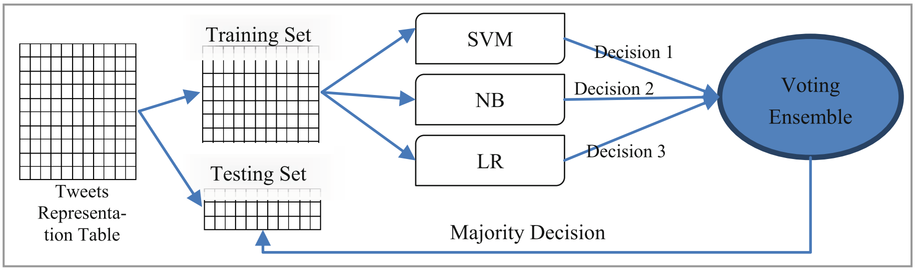


Fig. 3. Majority voting classifier ensemble

As shown in Fig. 3, the extracted features from the input tweets collection are split into two sets, training and testing set. The training set is passed to each classifier to register its decision. Then the final decision output from the voting ensemble is the majority decision obtained from the three classifiers. The voting ensemble model will consider this tweet as positive one because this is the majority decision. Finally, the testing set is used to validate the accuracy of the built classifier ensemble model.

4 Experimental Results and Discussion

The proposed system is implemented and its accuracy is measured against different well-known datasets in the area of twitter sentiment analysis. The preprocessing and feature extraction steps are implemented in Java with the support of Stanford CoreNLP library. The feature selection and the classification ensemble model are implemented using RapidMiner® tool. All the experiments were conducted using a machine with Intel® Core^{TM} i7-3770 CPU @ 3.40 GHz and 8.00 GB memory, running 64-bit Windows 7 Enterprise Edition®.

4.1 Datasets

Four datasets are used in evaluating the designed experiments in order to evaluate the performance of the proposed system. The distribution of the positive and negative polarities in the used datasets is shown in Table 1.

Table 1. Distribution of positive and negative polarities in the datasets

Dataset	Number of tweets	
	Positive	Negative
Stanford-1K	500	500
Stanford-3K	1500	1500
Sanders	201	293
HCR	211	386

Stanford Twitter Sentiment Corpus contains about 1.6M tweets (800,000 positive and 800,000 negative tweets) collected by a scrapper that calls Twitter API for some queries [24]. In our experiments, we did not use the complete training dataset due to the computational limitations. We perform unified sampling to obtain two sample datasets, Stanford-1K and Stanford-3K with 1000 and 3000 tweets respectively.

The third data sets is called **Sanders Dataset** [25] which contains about 5513 hand classified tweets with four labels: positive, negative, neutral and irrelevant. Twitter API is used with four search terms: @apple, #google, #microsoft and #twitter. We could not obtain all these tweets because most of them are currently invalid or deleted. In our experiments, we are interested in positive and negative labeled tweets only, which are about 201 positive and 293 negative tweets.

The fourth dataset is called **Health Care Reform (HCR) Dataset**. This dataset is collected by crawling tweets that contain the hashtag “#hcr” in March 2010 [26]. Some of these tweets are manually labeled into positive, negative and neutral label. In our experiment, we are interested in positive and negative ones only, which are about 597 tweets (211 positive and 386 negative).

4.2 Sentiment Analysis Results

Prominent Features Set. The objective of this experiment is to answer the first research question and decide what are the prominent features set that achieves the highest accuracy. Table 2 reports the accuracy of the proposed system using the majority voting ensemble model. Each dataset is represented with the different combinations of the Bag-of-Words (BoW), Lexicon-based features (Lex), Emoticon-based features (Emo) and PoS features (PoS).

Table 2. Accuracy for different combinations of features sets

Features Set	Accuracy (%)			
	Stanford-1K	Stanford-3K	Sanders	HCR
BoW	73.90	76.00	93.53	84.58
BoW + Lex	77.90	76.53	93.73	83.91
BoW + Emo	74.50	75.27	93.33	**84.75**
BoW + PoS	74.00	75.60	93.53	84.41
BoW + Lex + Emo	**78.70**	76.57	93.73	**84.75**
BoW + Lex + PoS	77.30	**77.27**	**93.94**	84.41
BoW + Emo + PoS	74.40	75.63	93.34	84.58
BoW + Lex + Emo + PoS	**78.70**	77.00	93.53	**84.75**

As shown in Table 2, the difference in the accuracy between BoW and other different combinations of features sets is not huge. Also, we notice that using the feature set that includes all the features (BoW + Lex + Emo + PoS) leads to the better accuracy as in Stanford-1K and HCR datasets. In Stanford-3K and Sanders datasets, the accuracy of the proposed system when using the complete features set still very high with very small margin compared to the highest accuracy. This allows us to state that *Lex and PoS features are good additions to traditional BoW features while Emo features do not add much to the overall accuracy.*

Using Information Gain (IG). The aim of this experiment is to answer the second research question and examine the effect of using information gain technique on the proposed system in two aspects: the accuracy enhancement and the dimension reduction. In the first part of the experiment, the accuracy of the standalone classifiers (SVM, LR and NB) and the Majority Voting Ensemble (MVE) model are reported in two cases, without using the information gain and with it applied.

In the second part of the experiment, we are interested in measuring the reduction ratio obtained after using information gain technique, because IG technique is mainly used to select the features that better match the given classes. Table 3 shows the comparison of the reported accuracy for each dataset and the feature vector length before and after using the IG technique.

Table 3. Accuracy comparison after using IG with different classifiers

Dataset		Accuracy (%)				Feature vector length	
		SVM	LR	NB	MVE	# Features	Reduction (%)
Stanford-1K	Without IG	63.6	65.5	62.4	64.8	911	**61.14**
	With IG	**78.1**	**74.5**	**76.5**	**78.7**	557	
Stanford-3K	Without IG	66.73	63.33	61.37	65.37	2400	**46.75**
	With IG	**79.1**	**71.13**	**77.77**	**77.27**	1122	
Sanders	Without IG	80.77	79.57	79.35	81.79	1023	**76.44**
	With IG	**92.71**	**90.11**	**91.91**	**93.94**	782	
HCR	Without IG	72.7	65.17	67.85	69.86	1357	**69.49**
	With IG	**81.22**	**75.37**	**85.09**	**84.75**	943	

As shown in Table 3, it is clear that using information gain technique enhanced the accuracy of the proposed system in all used classifiers with about 15% on average. In addition, using IG technique reduces the feature vector length for each dataset with about 63.45% on average. This allows the classifiers to distinguish efficiently between positive and negative classes with lower computational requirements. From these results, we can conclude that using the information gain (IG) technique not only reduces the dimension of the feature vector greatly, but also enhances the performance of the model classifier very efficiently.

Majority Voting Ensemble (MVE) Evaluation. In this experiment, we target to evaluate the performance of the MVE model in order to answer the third research question. The performance of each classifier is measured for all the datasets and the parameters that achieved the highest accuracy are recorded. MVE model is also tested by combining the decisions of the standalone classifiers with the optimal parameters and its accuracy is also recorded to be compared with other algorithms as shown in Table 4.

Table 4. Accuracy comparison for different classifiers

Classifier	Best accuracy (%)			
	Stanford-1K	Stanford-3K	Sanders	HCR
SVM	78.10	**79.10**	92.71	81.22
LR	74.50	71.13	90.11	75.37
NB	76.50	77.77	91.91	**85.09**
MVE	**78.70**	77.27	**93.94**	84.75

As shown in Table 4, SVM and NB classifiers have good performance with respect to different datasets, while LR has the worst accuracy results. Regarding the Majority Voting Ensemble (MVE) model, its performance is affected by the performance of the individual classifiers. For example, in Stanford-1K and Sanders datasets, when the base classifiers have good results, MVE outperforms them and achieve better results with accuracy 78.70% and 93.94% respectively. In HCR dataset, LR achieves very low accuracy, about 75.37%, compared to SVM and NB results. We can see that MVE tries to recover from such drop and achieves about 84.75% accuracy with a little bit difference to the highest accuracy that achieved by NB classifier (about 85.09%).

From these results, we can notice that MVE achieves the highest accuracy, and even outperforms the individual classifiers, when the results of these classifiers are near. In the case that one classifier has soft performance; MVE tries to recover from such performance and achieves good results which are very near to the best ones.

5 Conclusions and Future Work

In this paper, we introduced an efficient system for Twitter sentiment analysis. The proposed system used different techniques to represent input labeled tweets with different features sets. The irrelevant and insignificant features are early pruned using Information Gain (IG) feature selection technique. The classifier ensemble is built from diversified set of base classifier to perform the classification task which is responsible for detecting the output sentiment polarity. Many experiments were conducted using the most commonly used tweets datasets to analyze the performance of the proposed system in different aspects.

The experimental results answered the three main research questions in this work. First, using IG feature selection technique boosted the accuracy of the individual classifiers and the ensemble model with about 15% on average. Second, the ensemble model tried to combine the performance of the base classifiers, but its results could be affected if one classifier was not suitable for the used dataset. Third, we can notice that the reported results of the lexicon-based features and PoS features enhanced the accuracy of the classifiers when added to the BoW features. On the other hand, emoticon-based features did not have this much addition.

As a future work, we may include the “neutral” tweets into the proposed system by adapting the feature extraction and classification steps to recognize these tweets efficiently. In addition, to support tweets from other languages, such as Arabic [27], the proposed system could be adapted to be multi-lingual system.

References

1. Gaur, M., Pruthi, J.: A survey on sentiment analysis and opinion mining. Int. J. Curr. Eng. Technol. **7**(2), 444–446 (2017)
2. Li, Y.-M., Li, T.-Y.: Deriving market intelligence from microblogs. Decis. Support Syst. **55**(1), 206–217 (2013)
3. Rui, H., Liu, Y., Whinston, A.: Whose and what chatter matters? The effect of tweets on movie sales. Decis. Support Syst. **55**(4), 863–870 (2013)
4. Kang, D., Park, Y.: Review-based measurement of customer satisfaction in mobile service: sentiment analysis and VIKOR approach. Expert Syst. Appl. **41**(4), 1041–1050 (2014)

5. Hassan, A., Abbasi, A., Zeng, D.: Twitter sentiment analysis: a bootstrap ensemble framework. In: The International Conference on Social Computing (SocialCom), Alexandria, VA (2013)
6. Manning, C., Raghvan, P., Schutze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
7. Lin, J., Kolcz, A.: Large-scale machine learning at twitter. In: The International Conference on Management of Data (SIGMOD 2012), New York, NY, USA (2012)
8. Vinodhini, G., Chandrasekaran, R.: Sentiment classification using principal component analysis based neural network model. In: The International Conference on Information Communication and Embedded Systems (ICICES 2014), Chennai, India (2014)
9. Ravi, K., Ravi, V.: A survey on opinion mining and sentiment analysis: tasks, approaches and applications. Knowl. Based Syst. **89**, 14–46 (2015)
10. Kharde, V., Sonawane, S.: Sentiment analysis of twitter data: a survey of techniques. Int. J. Comput. Appl. **139**(11), 5–15 (2016)
11. Ghiassi, M., Skinner, J., Zimbra, D.: Twitter brand sentiment analysis: a hybrid system using n-gram analysis and dynamic artificial neural network. Expert Syst. Appl. **40**, 6266–6282 (2013)
12. Agarwal, B., Mittal, N.: Prominent Feature Extraction for Sentiment Analysis. Socio-Affective Computing Series. Springer International Publishing (2016)
13. Agrawal, A., Xie, B., Vovsha, I., Rambow, O., Passonneau, P.: Sentiment analysis of twitter data. In: Proceedings of the Workshop on Languages in Social Media (LSM 2011), Stroudsburg, PA, USA (2011)
14. Zhang, L., Ghosh, R., Dekhil, M., Hsu, M., Liu, B.: Combining lexicon-based and learning-based methods for twitter sentiment analysis. HP Laboratories (2011)
15. Mohammad, S., Kiritchenko, S., Zhu, X.: NRC-Canada: building the state-of-the-art in sentiment analysis of tweets. In: Proceedings of the Seventh International Workshop on Semantic Evaluation Exercises (SemEval-2013), Atlanta, GA, USA (2013)
16. Saif, H., He, Y., Alani, H.: Semantic sentiment analysis of twitter. In: Proceedings of the 11th International Conference on the Semantic Web (ISWC 2012), Berlin, Heidelberg (2012)
17. Kuncheva, L.: Combining Pattern Classifiers: Methods and Algorithms, 2nd edn. Wiley, New York (2014)
18. Fumera, G., Roli, F.: A theoretical and experimental analysis of linear combiners for multiple classifier systems. IEEE Trans. Pattern Anal. Mach. Intell. **27**(6), 942–956 (2005)
19. Lin, J., Kolsz, A.: Large-scale machine learning at twitter. In: Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data (SIGMOD 2012), New York, NY, USA (2012)
20. Rodríguez-Penagos, C., Atserias, J., Codina-Filba, J., Garcıa-Narbona, D., Grivolla, J., Lambert, P., Saur, R.: FBM: combining lexicon-based ML and heuristics for social media polarities. In: Proceedings of the Seventh International Workshop on Semantic Evaluation, Atlanta, GA, USA (2013)
21. Clark, S., Wicentwoski, R.: SwatCS: combining simple classifiers with estimated accuracy. In: Proceedings of the Seventh International Workshop on Semantic Evaluation, Atlanta, GA, USA (2013)
22. Da Silva, N., Hruschka, E., Hruschka Jr., E.: Tweet sentiment analysis with classifier ensembles. Decis. Support Syst. **66**, 170–179 (2014)
23. Liu, B., Hu, M., Cheng, J.: Opinion observer: analyzing and comparing opinions on the web. In: 14th International World Wide Web Conference, Chiba, Japan (2005)
24. Stanford Twitter Sentiment Corpus. http://help.sentiment140.com/for-students. Accessed May 2017

25. Sanders Dataset. http://www.sananalytics.com/lab/. Accessed May 2017
26. Speriosu, M., Sudan, N., Upadhyay, S., Baldridge, J.: Twitter polarity classification with label propagation over lexical links and the follower graph. In: Proceedings of the First Workshop on Unsupervised Learning in NLP (EMNLP 2011), Stroudsburg, PA, USA (2011)
27. Mostafa, A.M.: An evaluation of sentiment analysis and classification algorithms for Arabic textual data. Int. J. Comput. Appl. **158**(2), 29–36 (2017)

Intelligent Systems and Applications

Development of an Ontology Based Solution for Energy Saving Through a Smart Home in the City of Adrar in Algeria

Djamel Saba[1,2](✉), Houssem Eddine Degha[2], Brahim Berbaoui[1], and Rachid Maouedj[1]

[1] Unité de Recherche en Energies Renouvelables en Milieu Saharien,
URER-MS, Centre de Développement des Energies Renouvelables,
CDER, 01000 Adrar, Algeria
saba_djamel@yahoo.fr, berbaoui.brahim@gmail.com,
ra_maouedj@yahoo.fr

[2] Laboratoire de l'Intelligence Artificielle et les Technologies de l'Information,
Faculté des Nouvelles Technologies de l'Information et de la Communication,
Université Kasdi Merbah Ouargla, 30000 Ouargla, Algeria
degha.houssem@outlook.com

Abstract. The exploitation of electrical equipment and the emergence of New Information and Communication Technologies (NICT), which led to a significant increase in electricity consumption. This article presents a decision-making tool for the choice of energy consumption, which makes it possible to ensure better energy efficiency. It examines how the semantic Web approach can be used to represent information about a resident, energy, behavior, and activities of residents. Ontology is the main element in this work because it represents information about concepts, properties, and relationships between concepts. This approach offers several advantages, such as sharing and reusing knowledge for good decision-making. We chose OWL (Ontology Web Language) for the formal representation of knowledge, SWRL rules (Semantic Web Rules) to present the intelligent reasoning of the solution, finally, the software Protégé2000 is used for the edition of knowledge. We applied our solution to a residence located in the city of Adrar in Algeria. This city is characterized by a climate and human activities specific to the region; these two characteristics have a great influence on the consumption of energy.

Keywords: Smart home · Energy efficient · Domain ontology
Web ontology language · Semantic web rule language · Protégé2000
Decision-making

1 Introduction

People in everyday life interact with the surrounding environment in many ways. They perceive the environment and act upon it. This reaction is related to the desire of people and the state of the internal environment of the house; furthermore, these reactions can reach manually. But with technological development that is very fast without forgetting

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 531–541, 2018.
https://doi.org/10.1007/978-3-319-74690-6_52

the large increase in population, it has become difficult or impossible to control all these interactions [1]. This requires the presence of a solution for this problem. In this perspective, we propose this work which concerns the intelligent management of a home. By definition, a smart home is an environment (family or work) with ambient intelligence and automatic control, which allows it to respond to the behavior of residents and provide them with various facilities. it comprises a set of geographically distributed sensors, computers, and actuators (pre-actuators) [2]. In the field of educational institutions, [3] presents peer-reviewed contributions on smart universities by various international research, design and development teams.

Intelligent environments have been the subject of research in recent years. [4, 5] present an intelligent monitoring solution to support the elderly and handicaps in this everyday life. Other research has been proposed to monitor and help people with Alzheimer's Disease. With the advent of new technologies, scientists are increasingly interested in the potential benefits they can bring to this population [6]. In the same context, another problem has been addressed that is related to the design and development of intelligent home-based wireless sensor network and real-time data fusion for determining the welfare of an elderly person living alone in a smart home [7]. Other work is specific to smart cities, among which, [8] proposes a study and tools improve road traffic, energy consumption, logistics, frameworks to provide new services and make decisions in a comprehensive way, assistance with driving, electric vehicles, public transport and surveys on the concepts of smart cities. [9] Presents a coherent and innovative vision of smart cities, built around a value architecture. It describes the limitations of the contemporary concept of Smart City and argues that the next stage of development must include not only physical infrastructure but also information technology and human infrastructure requiring the intensive integration of technical solutions the Internet of Thing (IOT) and social computing. However, with the increasing integration of digital technologies and the Internet in learning, the demand for intelligent learning has grown steadily, especially in smart city scenarios. As the need for lifelong learning increases, intelligent learning environments in cities should be equipped to respond to people's demands. Intelligent learning/education is also one of the key applications of smart cities. In this context, [10] presents a study on intelligent learning in China by providing comprehensive and accurate data from different contexts of intelligent learning. In particular, it is studying intelligent learning in smart cities, which extends the concept of intelligent learning to cover both formal and informal learning and to support lifelong learning.

Around our contribution, the city of Adrar is characterized by specific climatic conditions with a very hot climate in summer and a very cold climate in winter [11]. However, the daily life of Adrar residents is rather special, characterized by certain traditions and activities (exchange of visits between families, hours of work and rest, festivals). All these characteristics lead to a huge increase in energy consumption due to the random use of various electrical equipment. This entails a high cost in the electricity consumption bill. The average annual electricity consumption for a family of four in the city of Adrar is estimated at 80000 DA (Algerian Dinars), which is equivalent to

666.66 Euros [12]. This contribution provides a smart solution for the management of a residence, which aims to reduce electricity consumption. To do this, we will realize a generic ontology which is considered one of the sets of ontologies for different purposes. From each job, we try to keep the concepts used and to remove the others, and we end with the addition of the other concepts that are necessary to achieve our objectives. For the choice of ontology, the Semantic Web provides a model for sharing and reusing data. Indeed, the smart home will have objects that will communicate with each other and with users via data. We want to build this concept so that it can be reused. Finally, ontology is at the heart of semantic web models. An overview of renewable energy technology in Adrar-Algeria.

Algeria to launch several projects in the national territory, among them the projects that are located in the region of Adrar (see Table 1) [12]:

Table 1. The photovoltaic power stations installed in Adrar (June 2014, March 2016)

Central	Installed power (MW)	Launch date
Adrar	10	June 2014
Adrar	20	October 2015
Kabertene - Adrar	3	October 2015
Z. Kounta - Adrar	6	January 2016
Timimoune - Adrar	9	February 2016
Reggane - Adrar	5	January 2016
Aoulef - Adrar	5	March 2016
Total power installed	58	

2 The Design of Ontology

There are several methods of developing ontology [13, 14]. The proposed method is an extension of these methods. It consists of a set of steps that are interesting to achieve our goal.

1. To determine the perimeter, the objectives, and the existing work: the ontology of this system would focus on the development of a smart home located in the city of Adrar in Algeria, whose main goal is economy energy. In this research axis, we distinguish several works [15–17].
2. Presentation of interesting terms in the field concerned by the study, we quote Building (Represents Residence), Human (Represents Residents), Source (Sources of Energy). We will also use existing works to define concepts related to our ontology. An extract of the concepts selected with the associated research work is presented in Table 2.
3. Present the class properties, instances and relations; refer to Tables 3, 4, 5 and 6.

Table 2. An excerpt from the classes

Classes	Classes description	Paper (ontology source)
Building	Represents a residence. Is an abstract and common class between all ontology's	Is a common concept between all the ontologies of the various works?
Source	The energy sources	"A semantic representation of energy-related information in future smart homes" [18]
Human	Represents the residents	"UPOS: User Profile Ontology with Situation-Dependent Preferences Support" [19], "Ontology-based management of the telehealth smart home, dedicated to elderly in loss of cognitive autonomy" [20]
Activity	Represents the daily activities	"Combining Activity Recognition and AI Planning for Energy-Saving Offices" [21], "A Semantic Approach with Decision Support for Safety Service in Smart Home Management" [22]
Appliance	Represents equipment	"An Ontology-Based Reasoning Approach Towards" [23], "Towards an ontology framework for intelligent smart home management and energy saving" [24]

Table 3. Examples of properties for ontology classes

Data-type – Property	Description	Concepts
PlaceName	Represents the room name	Place
PlaceSize	Represents the surface of a room	
ApplianceName	Represents the equipment name	Appliance
AppliancePower	Represents the equipment power	
SourceName	The energy sources name	Source
SourcePower	Represents the energy production	
HumanFullname	The resident name	Human
HumanSex	The resident sex	
HumanEmail	Resident E-Mail address	

Table 4. Examples on the facets of the attributes

Attribute	Type	Concepts
BuildingAddress	Alphanumeric	Building
PlaceName	Alphabetical	Place
ApplianceName	Alphabetical	Appliance
SourceName	Alphabetical	Source
SourcePower	Digital	Source
ActivityName	Alphabetical	Activity

Table 5. Examples of relations

Relation	Description	Associated concepts
BuildingHasPlace	A residence comprises one or more room	Building, Place
HumanHasActivity	Each resident is characterized by his activities	Human, Activity
BuildingHasSource	A residence is powered by at least one energy source	Building, Source
PlaceHas Appliance	Each room (places) includes one or more appliances	Place, Appliance

Table 6. Examples of ontology instances

Instances	Data-type – Property	Classes
Photovoltaic, Wind	SourceName	Source
PlaceName	Kitchen, living room, bedroom	Building
PlaceSize	20 m^2, 25 m^2, 30 m^2	Building
ApplianceName	Economy Lamp, Air conditioning, Plasma TV on standby	Appliance
AppliancePower	25 W, 150 W, 5 W, 2000 W, 0.3 W, 100 W, 6 W	Appliance

3 Implementation of the Solution

3.1 Implementation in an Ontology Editor

The solution information (classes, properties, individuals, etc.) was manually edited in the Protégé 5 software (see Fig. 1).

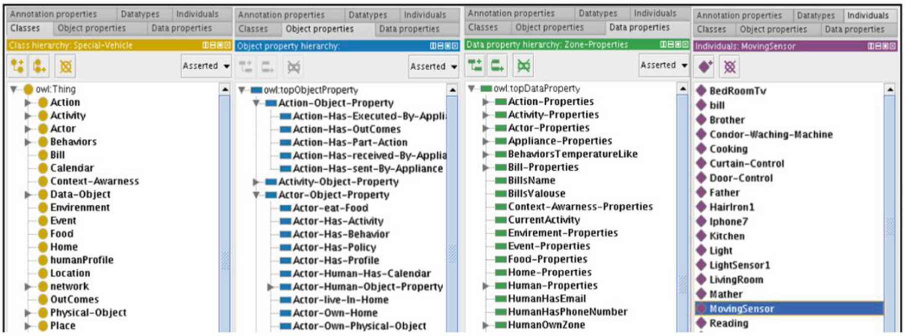


Fig. 1. Ontology elements in Protégé-OWL 5

3.2 Implementation of Intelligent Reasoning Rules

SWRL allows adding relationships, depending on the values for variables and the satisfaction of the rules. The antecedent and consequent of a rule are conjunctions of atoms. The atoms can have the following forms:

- C (x) where C is an OWL description, x is either a variable, an OWL individual, or OWL datavalues.
- P (x, y) where P is an OWL property (Object_property or data_type_property), x is either a variable or an OWL individual and y is either a variable, an OWL individual, or OWL data values.
- SameAs (x, y), differentFrom (x, y) where x and y are OWL variables or individuals.

The intelligent reasoning of the solution is based on a set of rules, in order to achieve the ultimate goal of energy saving. In this section, we will present some rules (see Fig. 2).

```
MovingSensor(?x) ∧  Devices _State(?x, ?stat) ∧ swrlb:equal(?stat, "true") ∧
Devices_Values(?x, ?val) ∧  swrlb:equal(?val, 0) ∧ DeviseLocateIn(?z, ?x) ∧
DeviseLocateIn(?z,   ?l)   ∧      Lights(?l)      ∧   swrlb:equal(?stat2,
Devices_State(?l, ?stat2) ∧ swrlb:equal(?stat2, "on") →Devices _State (?l,
"of")                               (R1)
```

The rule (R1) makes it possible to identify the presence of a person in a place (room) inside the house. If there is none, all the lamps connected in this room are switched off.

```
Illumination_Sensor(?x)  ∧  Device_location(?x,  ?loc)  ∧  swrlb:equal(?loc,
“indoor”) ∧ Device_Values(?x, ?val) ∧ (?val <= 20) ∧ illumination_Sensor(?y)
∧   Device_location(?y,   ?loc2)   ∧   swrlb:eqyal(?loc2,   “outdoor”   )   ∧
Device_Values(?y,    ?val    2)    ∧    swrlb:greaterThan(?val2?,70)    ∧
DeviceHasLocation(?x,?z)  ∧  DeviceHasLocation(?y,?z)    ∧  Light(?l)  ∧
DeviceHasLocation(?l,?z)   ∧   window(?w)   ∧   ThingHasLocation(?w,?z)   →
Device_Stat(?l,“of”) ∧ window_state(?w,”open”)                    (R2)
```

Ruler (R2) can open blinds when outdoor lighting is acceptable.

```
MovingSensor(?x) ∧  Devices_State(?x, ?stat) ∧  swrlb:equal(?stat, "true") ∧
Devices_Values(?x, ?val) ∧  swrlb:equal(?val, 0) ∧ DeviseLocateIn(?z , ?x) ∧
DeviseLocateIn(?z , ?l) ∧  Appliance(?l)  ∧  Devices_State(?l, ?stat 2) ∧
swrlb:equal(?stat2, "on") → Devices_State(?l, "of")”                 (R3)
```

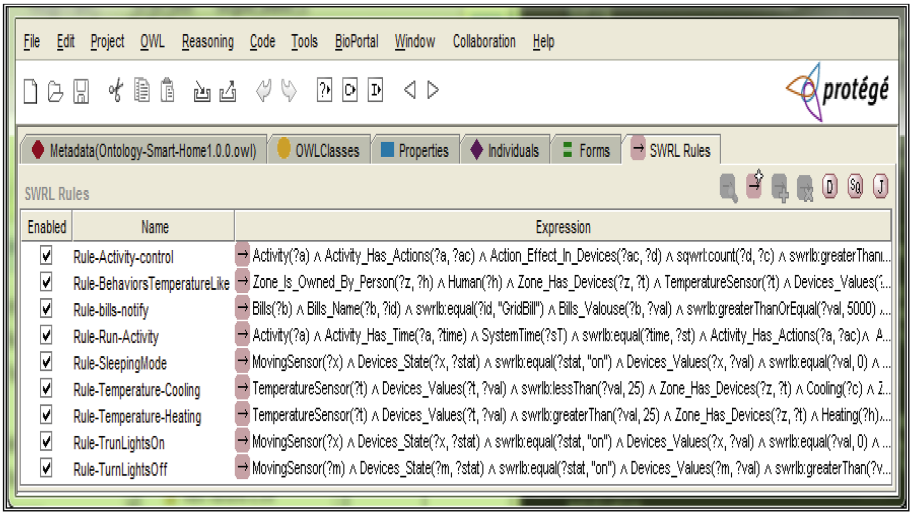
The rule (R3) allows turning off all equipment if the house does not contain anyone.

```
Temperature_sensor(?x)  ∧ Device_location  (?x,  ?AirCond)  ∧ Device_Stat( ?l,
,"on")   ∧   Temperature_values( ?x, ?val)    ∧   swrlb:  less  than(?val,25)   →
Device_Stat( ?l, ,"of")                                 (R4)
```

This rule turns off the air conditioner if the ambient temperature is below 25°.

```
Brightness_sensor(?x)   ∧  Device_location  (?x,  ?lights)  ∧  Device_Stat( ?l,
,"on") ∧ Brightness_values( ?x, ?val) ∧ swrlb: less than(?val,7000) ∧ swrlb:
greater than (?val,6000) → Device_Stat( ?l, ,"of")
                          (R5)
```

This rule makes it possible to extinguish the lamps of a room in the house if the illumination is strictly superior to 6000 and strictly inferior to 7000.



Enabled	Name	Expression
☑	Rule-Activity-control	→ Activity(?a) ∧ Activity_Has_Actions(?a, ?ac) ∧ Action_Effect_In_Devices(?ac, ?d) ∧ sqwrl:count(?d, ?c) ∧ swrlb:greaterThan...
☑	Rule-BehaviorsTemperatureLike	→ Zone_Is_Owned_By_Person(?z, ?h) ∧ Human(?h) ∧ Zone_Has_Devices(?z, ?t) ∧ TemperatureSensor(?t) ∧ Devices_Values(...
☑	Rule-bills-notify	→ Bills(?b) ∧ Bills_Name(?b, ?id) ∧ swrlb:equal(?id, "GridBill") ∧ Bills_Valouse(?b, ?val) ∧ swrlb:greaterThanOrEqual(?val, 5000) ...
☑	Rule-Run-Activity	→ Activity(?a) ∧ Activity_Has_Time(?a, ?time) ∧ SystemTime(?sT) ∧ swrlb:equal(?time, ?st) ∧ Activity_Has_Actions(?a, ?ac)∧ A...
☑	Rule-SleepingMode	→ MovingSensor(?x) ∧ Devices_State(?x, ?stat) ∧ swrlb:equal(?stat, "on") ∧ Devices_Values(?x, ?val) ∧ swrlb:equal(?val, 0) ∧ ...
☑	Rule-Temperature-Cooling	→ TemperatureSensor(?t) ∧ Devices_Values(?t, ?val) ∧ swrlb:lessThan(?val, 25) ∧ Zone_Has_Devices(?z, ?t) ∧ Cooling(?c) ∧ Z...
☑	Rule-Temperature-Heating	→ TemperatureSensor(?t) ∧ Devices_Values(?t, ?val) ∧ swrlb:greaterThan(?val, 25) ∧ Zone_Has_Devices(?z, ?t) ∧ Heating(?h)...
☑	Rule-TrunLightsOn	→ MovingSensor(?x) ∧ Devices_State(?x, ?stat) ∧ swrlb:equal(?stat, "on") ∧ Devices_Values(?x, ?val) ∧ swrlb:equal(?val, 0) ∧ ...
☑	Rule-TurnLightsOff	→ MovingSensor(?m) ∧ Devices_State(?m, ?stat) ∧ swrlb:equal(?stat, "on") ∧ Devices_Values(?m, ?val) ∧ swrlb:greaterThan(?v...

Fig. 2. Les règles de la solution éditées dans Protégé-OWL 5

4 Results and Discussion

To test the solution proposed, we chose a house located in the city of Adrar, Algeria. The town of Adrar in an Algeria is located in the south-west of the country. It is characterized by a desert climate, very hot especially in summer and short and very mild winters. The house concerned by the study includes a number of rooms (kitchen, living room, garage, …), each room includes a set of equipment (see Table 7). We are interested in electrical equipment. Finally, the family involved in the study includes four people (father, mother and two children).

Table 7. Examples of the rooms and its electrical equipment.

Room (area)	Electrical appliance	Number (n_{app})	Power (p_{app})
Kitchen	Light	2	100
	Refrigerator	1	150
	Electric water boiler	1	2000
	Ceiling fan (Diameter 122 cm)	1	55
	Air exchanger (exiting steam)	1	90
	Coffee maker	1	1160
Living room	Light	3	100
	TV (19″ colour)	1	40
	Parabolic demodulator	1	18
	Home internet router	5	1
	PlayStation 4 (PS4)	1	80
	Electric car	1	34000

4.1 Scenarios

To show the results of the solution, we propose two scenarios, without and with the intervention of the solution. We chose a typical day of the year, so we selected the day of August 20, 2017. With this scenario, we try to show the usual consumption of energy (see the Table 8). To calculate the energy consumed $(E_{app}(wh))$ by each equipment which is characterized by a nominal energy $(P_{app}(w))$ over a delay $(T_f(\text{minutes}))$, we will use the following formulas:

$$E_{app}(wh) = P_{app}x\frac{T_f}{60} \tag{1}$$

The total energy is calculated by the following formula:

$$E_{tot}(wh) = \sum\nolimits_{i=1}^{n} E_{appi} \tag{2}$$

Table 8. Excerpt from the standard scenario

Time	Actors	Activities	Actions	Switched-on	Energy consumed (Formula 1)
00:00–06:00	Husband, Wife, Child	Sleeping	Nothing	Air conditioner	54000
06:00–07:15	Husband, Child	Sleeping	Nothing	Air conditioner	11250
06:00–06:15	Wife	Weak up	Go to bathroom	Hall lights, Bathroom lights, Water boiler	11250
…	…	…	…	…	…
21:00–21:30	Husband	Prepare tomorrow thinks	Go to office	Air conditioner, Office lights, Laptop	4750
21:30–00:00	Husband, wife	Watch movie	Go to living room	TV, Parabolic demodulator	145

The second scenario with the intervention of the solution can intervene in many cases, such as forgetfulness or wrongdoing by people, as well as disruptions such as non-exploitation of external natural resources such as heat and solar light. We propose a solution to the energy economy that is mainly related to external natural resources (see Table 9).

Table 9. The temperature and the lighting that characterize the environment

Time	Temperature (°C)	Brightness value (Lux)	Brightness description
00:00–06:00	22	1366	Completely dark
06:00–06:15	23	4098	Illumination is weak
06:15–06:45	31	6830	Illumination is weak
…	…	…	…
21:00–21:30	25	0	Completely dark
22:00–00:00	25	0	Completely dark

Table 10, shows the energy consumption by the intervention of the proposed solution.

Table 10. The energy consumed by the intervention of the solution

Time	Energy consumed (scenario 1)	Energy consumed (scenario 2)	The rule used in scenario 2
00:00–06:00	54000	0	R4
06:00–07:15	11250	11250	/
06:00–06:15	625	550	R5
…	…	…	…
21:00–21:30	4750	4750	/
21:30–00:00	145	145	/

One of the reasons for the random consumption of electric power is the mistakes committed by the human person, such as forgetting a device that works, although not manipulated (leaving the human to the television or radio or computer working in spite of sleep). A second factor is the lack of interest in natural resources such as lighting and temperature in the outside of the house. In our work, we have taken advantage of these factors to achieve a significant energy economy. As examples, in Table 10, we have taken advantage of the proposed smart solution for a large economy by turning off the equipment (air conditioner from midnight to 6 am) through external heat exhaustion.

5 Conclusion and Perspectives

Due to the irrational management of energy resources (equipment and sources of energy), which has created a major problem to meet the growing demand for energy resources as well as huge bills. All these problems forced us to look for solutions.

In this article, we presented a new solution that concerns intelligent management in a house located in the city of Adrar in Algeria. We chose ontology as a knowledge representation approach because of the advantages that characterize this approach. This solution provides an automatic decision tool, it takes into account the climatic data of the environment of the house, the errors made by the people and it also supports the specificity of the region of Adrar in Algeria, as the traditions and the activities of people. Following the results presented in the previous section, we have benefited from high energy efficiency. Our contribution needs to be enriched through collaborative work between experts in the field of energy management. The next step is the development of a graphical interface for a good demonstration of the results. This solution must be tested to identify existing gaps.

References

1. Chun, C.: Interaction between human & building environment. Build. Environ. **88**, 1–2 (2015). https://doi.org/10.1016/j.buildenv.2015.01.004
2. Alaa, M., Zaidan, A.A., Zaidan, B.B., et al.: A review of smart home applications based on internet of things. J. Netw. Comput. Appl. (2017). https://doi.org/10.1016/j.jnca.2017.08.017
3. Uskov, V.L., Bakken, J.P., Howlett, R.J., Jain, L.C.: Smart Universities Concepts, Systems, and Technologies. Springer (2017)
4. Hussain, A., Wenbi, R., da Silva, A.L., et al.: Health and emergency-care platform for the elderly and disabled people in the smart city. J. Syst. Softw. **110**, 253–263 (2015). https://doi.org/10.1016/j.jss.2015.08.041
5. Wong, J.K.W., Leung, J., Skitmore, M., Buys, L.: Technical requirements of age-friendly smart home technologies in high-rise residential buildings: a system intelligence analytical approach. Autom. Constr. **73**, 12–19 (2017). https://doi.org/10.1016/j.autcon.2016.10.007
6. Paquette, G., Morakabati, M., Ménard. C., et al.: Les maisons intelligentes et le dépistage des troubles cognitifs : recension des écrits. In: 1er congrès québécois Rech. en Adapt, p. 64 (2015)
7. Suryadevara, N.K., Mukhopadhyay, S.C., Barrack, L.: Towards a smart non-invasive fluid loss measurement system. J. Med. Syst. (2015). https://doi.org/10.1007/s10916-015-0206-6
8. Alba, E., Chicano, F., Luque, G.: Smart Cities: First International Conference, Smart-CT 2016, 15–17 June 2016, Málaga, Spain, Proceedings (2016)
9. Dustdar, S., Nastić, S., Šćekić, O.: Smart Cities : The Internet of Things, People and Systems. Springer (2017)
10. Liu, D., Huang, R., Wosinski, M.: Smart Learning in Smart Cities (2017). https://doi.org/10.1007/978-981-10-4343-7
11. dzmeteo: Météo Adrar, Prévisions de 10 jours Adrar, Algérie Météo. 1 (2017)
12. sonelgaz: Ferme éolienne d'Adrar. 1 (2017)
13. Saba, D., Laallam, F.Z., Hadidi, A.E., Berbaoui, B.: Optimization of a multi-source system with renewable energy based on ontology. Energy Procedia **74**, 608–615 (2015). https://doi.org/10.1016/j.egypro.2015.07.787
14. Saba, D., Zohra Laallam, F., Belmili, H., et al.: Development of an ontology-based generic optimisation tool for the design of hybrid energy systems. Int. J. Comput. Appl. Technol. (2017). https://doi.org/10.1504/ijcat.2017.084773

15. Hendler, J., Berners-Lee, T.: From the semantic web to social machines: a research challenge for AI on the World Wide Web. Artif. Intell. **174**, 156–161 (2010). https://doi.org/10.1016/j.artint.2009.11.010
16. Chavarriaga, E., Jurado, F., Díez, F.: An approach to build XML-based domain specific languages solutions for client-side web applications. Comput. Lang. Syst. Struct. (2017). https://doi.org/10.1016/j.cl.2017.04.002
17. Harrington, J.L., Harrington, J.L.: XML support. In: Relational Database Design and Implementation, Chap. 26, pp. 523–541 (2016)
18. Kofler, M.J., Reinisch, C., Kastner, W.: A semantic representation of energy-related information in future smart homes. Energy Build. (2012) https://doi.org/10.1016/j.enbuild.2011.11.044
19. Sutterer, M., Droegehorn, O., David, K.: UPOS: user profile ontology with situation-dependent preferences support. In: Proceedings of 1st International Conference on Advances in Computer-Human Interaction ACHI 2008 (2008). https://doi.org/10.1109/achi.2008.23
20. Latfi, F., Lefebvre, B., Descheneaux, C.: Ontology-based management of the telehealth smart home, dedicated to elderly in loss of cognitive autonomy. In: CEUR Workshop Proceedings (2007)
21. Georgievski, I., Nguyen, T.A., Aiello, M.: Combining activity recognition and AI planning for energy-saving offices. In: 2013 IEEE 10th International Conference on Ubiquitous Intelligence and Computing, 2013 IEEE 10th International Conference on Autonomic and Trusted Computing, pp. 238–245. IEEE, (2013)
22. Huang, X., Yi, J., Zhu, X., Chen, S.: A semantic approach with decision support for safety service in smart home management. Sensors **16**, 1224 (2016). https://doi.org/10.3390/s16081224
23. Kim, Y., Yoo, S.Y., Cheong, Y., et al.: An ontology-based reasoning approach towards. In: 2011 IEEE Consumer Communication and Networking Conference, pp. 850–854. IEEE (2011)
24. Grassi, M., Nucci, M., Piazza, F.: Towards an ontology framework for intelligent smart home management and energy saving. In: 2011 IEEE International Symposium Industrial Electronics, pp. 1753–1758. IEEE (2011)

Embeddings of Categorical Variables for Sequential Data in Fraud Context

Yoan Russac[1](✉), Olivier Caelen[2,3], and Liyun He-Guelton[2,3]

[1] ENSAE ParisTech, Paris, France
yoan.russac@ensae-paristech.com
[2] R&D Worldline, Brussels, Belgium
{olivier.caelen,liyun.he-guelton}@worldline.com
[3] R&D Worldline, Lyon, France

Abstract. In this paper we propose a new generic method to work with categorical variables in case of sequential data. Our main contributions are: (1) The use of *unsupervised* methods to extract sequential information, (2) The generation of embeddings including this sequential information for categorical variables using the well-known Word2Vec neural network. The use of embeddings not only reduced the memory usage but also improved the machine learning algorithms learning capacity from data compared with commonly used One-Hot encoding. We implemented those processes on a real world credit card fraud dataset, which represents more than 400 million transactions over a one year time window. We demonstrated that we were able to reduce the memory usage by 50% and to improve performance by 3% points while using only a small subset of features.

Keywords: Categorical variable · Word2Vec · Embeddings
Credit card fraud detection · Machine learning

1 Introduction

In a steadily growing e-commerce market with an overall loss due to fraudsters of 6.97 cents per every $100 in 2016, detecting fraudulent pattern is of great importance. Fraud data possesses particularities: unbalanced classes, concept drift, large amount of data, difficulty to define a cost function, overlapping, etc. [1]. For those reasons, fraud detection has gained popularity amongst the machine learning community during the past decade [1,9].

Another difficulty in fraud detection is that categorical variables are over-represented with more than 80% of our total variables. These variables vary amongst others, including the country of the transaction, the merchant type, the currency used, etc. If we do not find a correct representation of these variables, the machine learning algorithms will not be effective. A standard way to

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 542–552, 2018.
https://doi.org/10.1007/978-3-319-74690-6_53

deal with a categorical variable is One-Hot encoding. However, One-Hot vectors have two main shortcomings: (1) They are high-dimensional and sparse. (2) The relations between different values of categorical variables is ignored. It was demonstrated in [2] that using a *supervised* method to create embeddings for categorical variables reduced the memory usage and improved the performance of the neural network as it gave a better data representation. Here, we propose an *unsupervised* method for our sequential fraud data. The advantage of *unsupervised* based methods is their flexibility for injecting different semantic knowledges such as external sources [11]. In our case, we use this approach to inject the sequential pattern of the transactions.

We proceed the following way: we create sequences centered on the different cardholders. Once we have a set of sequences for different cardholders, we use an internal process for generating the embeddings with Word2Vec [5,6]. Word2Vec is a neural network frequently used for Natural Language Processing (NLP) tasks [7,10].

This paper is organized as follows: in Sect. 2 we describe our approach, in Sect. 3 we describe the credit card dataset and the experiment we did. Finally Sect. 4 concludes with an outlook and our future work.

2 Approach

The Word2Vec neural network is often used to learn linguistic regularities by representing words with vectors. The ordinary usage of this neural net is to feed it with sentences from a corpus of documents and to collect the embedding vectors after a training phase. Our process is very similar. The equivalent of the sentences will be variable sequences from our fraud dataset. These sequences will then be used to train the Word2Vec neural network. In this section, we will describe our approach: how we obtained the sequences and how we generated the embeddings using Word2Vec.

2.1 Sequences Generation

There are two ways of generating the sequences. We can either generate sequences for each categorical variable or for joint variables. The joint variable can take into account the interactions between different categorical variables in addition to the sequential information.

Univariate Sequences Generation: In the first case, to create our sequences we decide to group the entire training set by cardholders. For example if a given cardholder buys during the days corresponding to the training set, in France, then in Belgium, and after that back in France, the associated sentence will be ‘France Belgium France’. We proceed the same way with all the cardholders in our training set. After this process we have a collection of sentences in which each sentence is linked to the list of transactions of a particular cardholder. This process is repeated for every categorical variable.

Multi-variate Sequences Generation: In a fraud context the joint information of the country of the transaction and the merchant type really matters. Variables used to monitor suspicious activity include this information. When using One-Hot encoding there is no easy way to have this sequential information. With two categorical variables we could decide to create their product, where for example 'FRA' and merchant_type = '1010' will give us 'FRA1010'. Once this new interaction variable is created, it is possible to use the previous procedure to create the embeddings for the joint variable. By doing so we could create any interaction variables between categorical variables and extract its sequential information through the use of our *unsupervised* embeddings.

The One-Hot encoding only includes the information of the current transaction. Thus our generation procedure is meaningful, because it allows to extract some of the information of the sequences of transactions. This can help detecting uncommon behaviors.

2.2 Word2Vec

Generally the embeddings extracted from Word2Vec reflect the linguistic regularities and semantic information of the input word sequences. Here, we are not interested in the semantic information but in the relative positions of words in the sequences. The outputs of the Word2Vec tool are the embedding vectors. A fundamental notion to understand the network is the context window. The context window represents the words surrounding a given word in a sentence. It contains past words but also future words. When using Word2Vec one has to set the size of the context window and the dimension of the embeddings to create the right neural architecture.

The neural network: Word2Vec contains 3 layers. The input layer, one hidden layer and the output layer. It is worth mentioning that two predictive methods are possible when using Word2Vec. The first one is skip-gram architecture, where context words are predicted given an input word. The second one is the CBOW (continuous bags-of-words) where we try to predict the appropriate word given a context window. We will focus on the CBOW predictive method. The significant part of the network is the weights between the hidden layer and the output layer. These weights are updated during the back propagation. At the end of the update the embedding vectors will be the weight matrix. The structure of the neural network is represented in the Fig. 1 where V denotes the cardinal of the vocabulary (i.e. the different words in the corpus) and N denotes the dimension of the embedding vectors (i.e. the hidden layer has N dimensions). With a CBOW structure the matrix of interest is $\widetilde{W}$. In the input layer, which contains C parts, where C is the context window's size, each one of the input words is one-hot encoded, thus each input block has a V-dimension. During the training phase the weights of the matrix $\widetilde{W}$ will be updated and after a sufficient number of iterations over the corpus, the embedding vectors are available.

In the Fig. 1 there are C blocks corresponding to a C-size context window. The outputs of neural the net are probabilities.

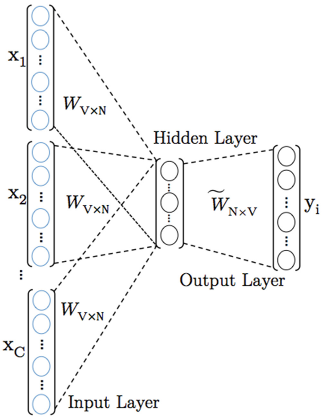


Fig. 1. Architecture of Word2Vec with CBOW technique

Subsampling frequent words: In a large corpus, stop-words can be really frequent. Observing co-occurrence of a frequent word and another word will bring less information. To counter this imbalance between occurrences of words in the corpus, a discard probability taking into account the frequency of the word was introduced in [6]. The more frequent a word in the sentences, the higher the probability of being discarded. In our dataset, the country Belgium was overrepresented, because our data are coming from a Belgian payment processor. Hence we used undersampling technique.

Negative sampling: The idea behind negative sampling is to update only a sample of the output vectors at each iteration instead of updating the entire weight matrix. Negative words (which are not the correct words given the context window) are sampled given a specific probability distribution and we only update the weights for those words.

These two extensions not only significantly reduced the computational requirements of the training process but also improved the quality of the embedding vectors [6].

3 Experiment

In this section we present our experiments. We compare the prediction's quality of several data structures: (1) One-Hot for every categorical variable, (2) Embedding for every categorical variable, (3) One-Hot and Embedding for every categorical variable (OHE in the different figures). We used both Logistic Regression and Random Forests to assess the precision.

3.1 Data and Experimental Design

Data: We worked on a labeled dataset provided by Worldline company. This set contained one year of transactions. Each day was composed of approximately

500 000 transactions. We decided to keep 3 categorical variables: merchant type, country of the transaction, local currency and a quantitative variable which is the amount of the transaction. Specialists in the fraud detection domain use much more variables [1] but our aim was to compare the quality of the predictions. Please note that transactions must be separated in two categories: e-commerce transactions which take place on the Internet and physical transactions in other words face-to-face ones. For example if a given cardholder makes face-to-face transactions in America and in Belgium in a 3 h time window there is probably something uncommon happening. However it is possible with e-commerce transactions. We generated different embedding vectors in function of the e-commercial status of the transaction.

Experimental Design: Our experiment is structured the following way: (1) Randomly select an initial date. Create the corresponding training and testing set. (2) Files processing along the required structure (One-Hot, embeddings). (3) Apply the algorithm on this processed training and testing files. The process was repeated 100 times and is illustrated with the following algorithm.

Algorithm 1. Experimental Design

```
Require: D: dataset, Processing method, L: learning algorithm
Prediction_list ← [ ]
$d_1, .., d_{100} \sim Sample(100, length(D))$        ▷ Select 100 initial dates
for $d_n \in \{d_1, ..., d_{100}\}$ do
    $(Training\_set, Testing\_set) \leftarrow ((d_n, d_{n+1}, d_{n+2}), d_{n+3})$
    $Training\_set \leftarrow Processing\ method(Training\_set)$
    $Model \leftarrow L(Training\_set)$
    $Prediction\_list \leftarrow Prediction\_list + [Model.fit(Testing\_set)]$
end for
return Prediction_list        ▷ Predictions for the 100 testing files
```

3.2 Treatment of Categorical Variable

We used the Python's *gensim*[1] package [8] to generate the embeddings. It has a very efficient implementation of Word2Vec. We created a list of list containing all the sentences from users in the 3 days training set. The sequences were obtained following the method presented in Subsect. 2.1. For every categorical variable, we then had to train the Word2Vec neural network feeding it with the associated sequence. For a given categorical variable we built a dictionary where every level of the variable was mapped to the associated embedding vector. We were then able to integrate these embedding vectors in a *Pandas*[2] dataframe.

As explained in the Subsect. 2.2 several parameters could be set for the configuration of Word2Vec. In our experiment we have chosen a context window

[1] https://radimrehurek.com/gensim/.

[2] http://pandas.pydata.org/.

of size 5, which means that in best case scenario the five previous words and the five following words would be considered. We chose the CBOW predictive technique because we empirically noticed that it outperformed the Skip-Gram architecture for the experiment. Negative sampling was used with 5 *noise words* per training. The threshold for subsampling frequent words was set to 10^{-3}. The embeddings were generated with this configuration. Several embeddings dimensions were tested (see Table 1).

We also created an extended model where we computed the embeddings of the variable Country × Merchant Type. It is another advantage of embeddings which allows to create the interaction between several categorical features.

3.3 Resampling Method

Before applying a Logistic Regression or Random Forests on the different training sets we had to deal with this unbalanced distributions issue (less than 0.2% of our entire dataset are fraudulent transactions). Machine learning algorithms usually perform poorly on such a training set [3]. We adopted a strategy similar to the *EasyEnsemble* strategy [4]. *EasyEnsemble* learns different aspects of the original majority class in an unsupervised manner. This is done by creating different balanced training sets by *undersampling*, learning a model for each dataset and then combining all predictions as in bagging. For a given training set our method was the following: (1) Collect every fraudulent transaction. (2) Select randomly a given number of non fraudulent transactions in the training set so that the proportion of non fraudulent transaction vs. fraudulent transaction in the new artificially created dataset will be 80%. (3) Create a model with this new dataset. We had to repeat (1–3) at least 100 times. To obtain a prediction on a new testing example, we took the average of the predicted probabilities of all the models created.

3.4 Performance Measure

Fraud classification problems often show very unbalanced classes. Therefore classical performance measures are not suitable. With an overall proportion of 0.2% of frauds, classifying every transaction as a legitimate one gave an accuracy (i.e. proportion of correct classification) of 99.8%, even if the model was absolutely naive. We also aimed at using realistic measures that made sense for real world fraud detection. Fraud experts check manually only hundreds of transactions (depending on the size of the team). Therefore we used the precision at k ($p@k$) which is the proportion of real frauds among the k riskiest transactions according to the chosen algorithm [1]. For example if $p@100 = 20\%$ then among the 100 riskiest transaction for the model, 20 of them were truly frauds.

The $p@k$ measure was somewhat variable, thus we wanted to use a more global metric to compare the models. Using the average of the K first transactions $p@k$ made sense in order to measure the global performance. Therefore we used the following metric:

$$Average\ p@K = \frac{1}{K}\sum_{i=1}^{K} p@i \tag{1}$$

As already discussed in [1] there was no need to observe the quality of the model beyond K transactions with this metric because fraud experts were unlikely to manually check more than K transactions in a single day. In our experiment, we fixed K at 500.

3.5 Results

To sum up the experiment, we run the model on 100 different 4-day-periods. For each one of those periods, we use a resampling method. This resampling method consists in building 100 new datasets which are extracted from the training set and where classes are much more balanced.

When we created embedding vectors we manually chose their dimensions (N). As far as we know, there is no theory to know which dimension one should consider. In practice we chose them empirically. On average, the more levels a categorical variable had the larger the dimension we took because we assumed that higher dimensions would allow us to gather more precise information on the sequences of transaction. In our data the country variable had 180 levels, the merchant type had more than 600 levels and the local currency had 150 levels. We created four different sizes that are gathered in the Table 1. The interaction variable is only included when it is specified that the model is *extended*.

Table 1. Embedding configurations (N) of the experiment

Variables	Country	Merchant type	Local currency	Country × Merchant type
Low dim	10	25	25	50
Med. dim	25	50	50	NA
Large dim	50	75	75	NA
Very large dim	80	150	150	NA

Logistic Regression: We implemented a logistic regression using the Python's package *scikit-learn*[3] and compared the results with One-Hot encoding versus other configurations. The results are reported on the Figs. 2 and 3. On the Fig. 2 we represented the Average p@500 and their $t - based$ confidence intervals ($\alpha = 0.05$). It shows that with small or medium dimensions using only embeddings was less effective than the classic One-Hot encoding but higher dimension improves the performance. Combining One-Hot encoding and embeddings gave an Average p@500 of 8.2% which must be compared to the 5.8% Average p@500 of the One-Hot encoding. Our method improved from 2.4% points the results in a Logistic Regression configuration and was statistically better than One-Hot.

[3] http://scikit-learn.org/stable/.

Three main conclusions to be drawn: (1) On average, the higher the embedding dimension, the better the results. But the difference was not significant with only 3 variables. (2) Embedding vectors gave slightly better results than One-Hot encoding but with a use of far less memory. In our example the memory usage of the dataframe containing 3 days of data with embedding vectors with small dimensions was 2 times less than with One-Hot. (3) Combining One-Hot and embedding techniques gave us better results than using One-Hot only. The p@100 gained 3% points to 4% points when we combined the techniques and thus enriched the data.

Random Forests: We implemented a Random Forests algorithm using *scikit-learn* and compared the results with One-Hot encoding versus other configurations. We reported the Average p@500 for the different methods on the Fig. 4. With a non-linear algorithm the results were different. (1) The Average p@500 was much higher than with a Logistic Regression. (2) The previous remark concerning the dimensions of the embeddings seemed infringed with this algorithm. (3) Adding One-Hot to the embeddings was less effective, but embeddings alone performed significantly better than One-Hot encoding according to this metric. The Average p@500 when I use One-Hot encoding was less than 37.5% and it was around 40% when using embeddings. (4) When we observed the 95% confidence intervals (Fig. 4) we could assure that we have a significant improvement when using embeddings.

The best model for the different configurations are represented in the Fig. 5.

Remark: The confidence intervals are larger than with the Logistic Regression algorithm because the $p@k$ vary much more with Random Forests than with Logistic Regression (as can be seen on Figs. 3 and 5)

On the Table 2 we reported $p@k$ for different values of k when using Random Forest with small embeddings configuration on the *extended* model. The *extended* model slightly outperformed the plain one. Different k values in the table demonstrated that the integration of the joint variable was a moderate success.

The conclusions for the Random Forest algorithm were: (1) Using non linear algorithm to detect fraudulent pattern performed well because p@100 rose from 9% in the best case scenario when using Logistic Regression to almost 48% with Random Forest. (2) Replacing the One-Hot encoding worked: e.g. the p@100 performance increase from more than 3% points. (3) As can be seen in the

Table 2. Comparison between *extended* model and non *extended* one

	Small embeddings	Small extended embeddings
p@50	55.31%	55.33%
p@150	43.27%	43.63%
p@250	37.48 %	37.73%
p@500	28.5%	28.72%

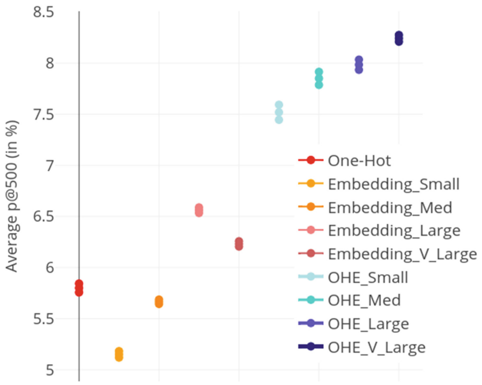


Fig. 2. *Average* $p@500$ and confidence interval using Logistic Regression (plot order follows the legend)

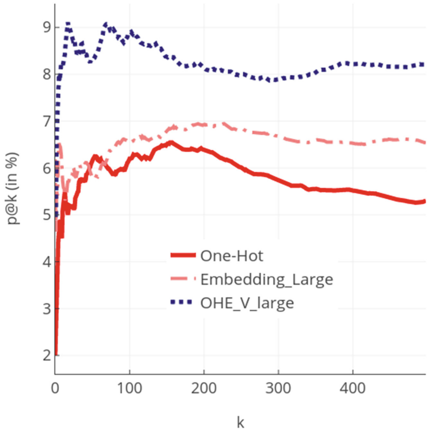


Fig. 3. $p@k$ in function of k with different encoding methods using Logistic Regression

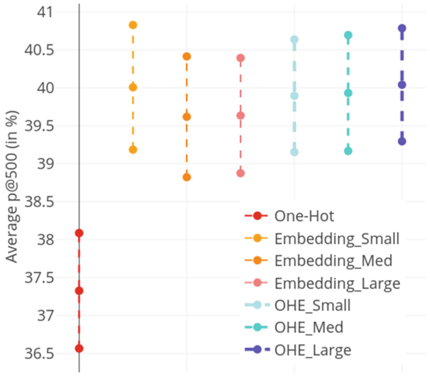


Fig. 4. *Average* $p@500$ and confidence interval using Random Forest (plot order follows the legend)

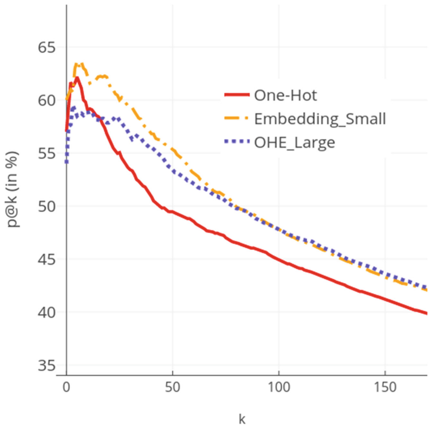


Fig. 5. $p@k$ in function of k with different encoding methods using a Random Forest algorithm

Table 2 using the extended model with the joint variable improved slightly the performance for different $p@k$ measures. In average we achieved a 0.2% points improvement by adding the joint variable.

4 Conclusions

In this paper we made the following contributions: (1) We implemented a generic method to generate *unsupervised* embeddings for categorical variables and applied it to credit card fraud detection. This method can be used for any sequential data and allows to inject sequential information through embeddings. (2) We created the embeddings in two different ways: for each categorical variable independently and for joint variables. By doing so we enriched the dataset. (3) Combining One-Hot encoding and embeddings improved the p@100 performance, which is one of Worldline's performance metric, by 3% points for both algorithms. (4) When using low dimensional embeddings we divided the memory usage by 2 in comparison to One-Hot encoding. We proved that the fraud detection performance (Average p@500) was enhanced with the support of the 95% confidence level.

Due to the time constraint and heavy computation charge, we only tested our approach with 3 categorical variables and a small time window. In the future, we will extend our experiments with all categorical variables (≈20 variables) and a larger time window. We also plan on implementing these methods in a *online learning* context and integrating the embeddings in an advanced machine learning algorithm, such as *LSTM* neural network to verify its efficiency.

References

1. Dal Pozzolo, A., Caelen, O., Le Borgne, Y.-A., Waterschoot, S., Bontempi, G.: Learned lessons in credit card fraud detection from a practitioner perspective. Expert Syst. Appl. **41**(10), 4915–4928 (2014)
2. Guo, C., Berkhahn, F.: Entity embeddings of categorical variables. CoRR, abs/1604.06737 (2016)
3. Japkowicz, N., Stephen, S.: The class imbalance problem: a systematic study. Intell. Data Anal. **6**(5), 429–449 (2002)
4. Liu, X.-Y., Wu, J., Zhou, Z.-H.: Exploratory undersampling for class-imbalance learning. IEEE Trans. Syst. Man Cybern. B (Cybernetics) **39**(2), 539–550 (2009)
5. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: Proceedings of Workshop at ICLR 2013, January 2013
6. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proceedings of the 26th International Conference on Neural Information Processing Systems, NIPS 2013, USA, vol. 2, pp. 3111–3119. Curran Associates Inc (2013)
7. Musto, C., Semeraro, G., de Gemmis, M., Lops, P.: Word embedding techniques for content-based recommender systems: an empirical evaluation. In: Castells, P. (ed.) RecSys Posters, CEUR Workshop Proceedings, vol. 1441 (2015). http://ceur-ws.org/
8. Rehurek, R., Sojka, P.: Software framework for topic modelling with large corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks. Citeseer (2010)
9. Trivedi, I., Monik, M., Mridushi, M.: Review of web crawlers with specification and working. Int. J. Adv. Res. Comput. Commun. Eng. **5**(1), 39–42 (2016)
10. Wen, Y., Yuan, H., Zhang, P.: Research on keyword extraction based on word2vec weighted textrank. In: 2016 2nd IEEE International Conference on Computer and Communications (ICCC), pp. 2109–2113, October 2016

11. Ziegler, K., Caelen, O., Garchery, M., Granitzer, M., He-Guelton, L., Jurgovsky, J., Portier, P.-E., Zwicklbauer, S.: Injecting semantic background knowledge into neural networks using graph embeddings. In: 2017 IEEE 26th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), pp. 200–205. IEEE (2017)

Hybrid Information Filtering Engine for Personalized Job Recommender System

Islam A. Heggo(✉) and Nashwa Abdelbaki

School of Communication and Information Technology, Nile University, Giza, Egypt
{i.heggo,nabdelbaki}@nu.edu.eg

Abstract. The recommendation system, also known as recommender system or recommendation engine/platform, is considered as an interdisciplinary field. It uses the techniques of more than one field. Recommender system inherits approaches from all of machine learning, data mining, information retrieval, information filtering and human-computer interaction. In this paper, we propose our value-added architecture of the hybrid information filtering engine for job recommender system (HIFE-JRS). We discuss our developed system's components to filter the most relevant information and produce the most personalized content to each user. The basic idea of recommender systems is to recommend items for users to suit their interests. Similarly the project tends to recommend relevant jobs for job-seekers by utilizing the concepts of recommender systems, information retrieval and data mining. The project solves the problem of flooding job-seekers with thousands of irrelevant jobs which is a frustrating and time-wasting process to let job-seekers rely on their limited searching abilities to dig into tons of jobs for finding the right job.

Keywords: Information retrieval · Hybrid recommender system · E-recruitment
Search engine · Ranking · Personalization · Domain-knowledge

1 Introduction

Traditionally recommender system was used to be employed in e-commerce applications. For example, Amazon finds for users the most products and items they would like. Instead of complicating the process to find interesting products, the e-commerce system will intelligently recommend interesting products for user.

Eventually recommendation systems become core model in different applications. The concept becomes popular in many categories such as recommending movies, music, news [1], jobs, friends, courses, restaurants, and more.

The challenge is how to recommend the most suitable jobs for job-seekers. This problem is considered as one of the recent important problems in the field of recommendation systems, data mining and information retrieval. Everything moves towards the online services and Internet. Nowadays online recruitment platforms become an essential channel for both recruiters and job-seekers. The idea is how to match job-seeker with job. Solving this issue of an online recruitment platform is substantial to achieve the highest level of satisfaction at both sides.

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 553–563, 2018.
https://doi.org/10.1007/978-3-319-74690-6_54

On the job-seeker side, job-seekers need to find an online platform that understands their preferences and behaviors to recommend the most interesting jobs. It is important to avoid annoying job-seekers with irrelevant jobs and emails which could be considered as spammy platform. On the other side, it is important also for recruiters to find only the suitable calibers who are matched to the vacancy they offer instead of screening all the applicants whose count could be in thousands [2].

This paper enumerates the various components to build a recommendation engine. It begins with different techniques of recommending items, the influential data to produce relevant jobs, the filtration of jobs based on inputs, the ranking of the recommend results and finally the different online and offline evaluation metrics for evaluating similar engines.

2 Literature Survey

Recommender systems are divided into four main categories. Content-Based Recommendation (CBR) is finding the similarity between the content of two profiles. Sometimes it is called cognitive filtering. It could be considered as an information retrieval problem. It mainly depends on analyzing the keywords and terms found in item profile and another item profile to calculate how they are similar to each other. Collaborative Filtering (CF) is a well-known technique which has proven its quality. The fundamental assumption of collaborative filtering is that if users X and Y rate n items similarly, or have similar behaviors (e.g. buying, watching, listening), hence they will rate or act on other items similarly [3–8]. Demographic-Based Recommendation (DBR) depends mainly on the demographic data of users (e.g. age, gender). Based on these data it works on clustering users into groups. Then it starts to handle all users among the same group equally [5, 6]. Knowledge-based Recommendation (KBR) depends on patterns and rules extracted from exploiting deep knowledge of the interesting items for users. That can be done through deep understanding of the market and application context, or through capturing users' preferences via dialogs or questions in some applications, and then the recommendation system considers these preferences to build its own discrimination tree of item attributes [3–8]. We witnessed the usage of CBR, CF and KBR approaches by CareerBuilder.com from United Stated. Proactiverecruitment.co.uk from UK relied mainly on CBR [9]. Prospects.ac.uk from UK and eRecruiter from Austria depend mainly on CBR and KBR. The leading job search engines LinkedIn and Glassdoor mainly rely on search-based KBR in addition to CF in some contexts.

3 Proposed Engine

Our system is a hybrid recommender system. It is based on the four above mentioned methodologies in addition to our developed modules which are behavioral-based recommendation (BBR), concept-based recommendation (COBR) and ontology-based recommendation (OBR). The final output of these modules is formed eventually in the form of textual search quires and search cut-off filters to hit our job search engine and retrieve matching jobs from our job index. Figure 1 shows our hybrid engine modules through a simplified flowchart.

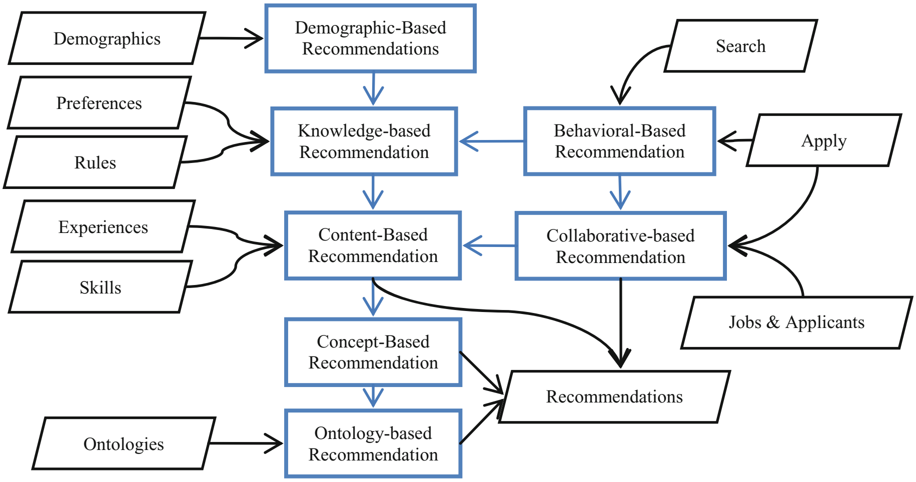

Demographics
Demographic-Based Recommendations
Search
Preferences
Rules
Knowledge-based Recommendation
Behavioral-Based Recommendation
Apply
Experiences
Skills
Content-Based Recommendation
Collaborative-based Recommendation
Jobs & Applicants
Concept-Based Recommendation
Recommendations
Ontologies
Ontology-based Recommendation

Fig. 1. Modules of our job hybrid recommendation engine

3.1 Hybrid Information Filtering Recommendation Algorithm

Content-Based and Knowledge-Based start by retrieving the jobs which match the content of the job-seeker profile based on tf-idf concepts (term frequency - inverse document frequency). We discussed this layer particularly in another publication.

Behavioral-based recommendation tracks the activities of job-seeker, it learns from user's previous actions. For instance, it track the jobs that the job-seeker applied to, then it calculate the most recent and frequent job-titles of these jobs to utilize them in the recommendations even if these job-titles were not present in the job-seeker's profile. Assume the job-seeker's profile includes only *backend developer*, but he applied frequently to *php developer*. Then Behavioral-based recommendations will consider that the *php developer* jobs are interesting for that job-seeker.

Concept-based recommendation tries to augment our search queries by using the job-seeker's profile content (content-based recommendations) to retrieve the recommended jobs. Then it extracts the most frequent keywords found in these content-based recommended jobs to augment the recommendation query and retrieve the conceptually similar jobs. As example, most of the recommended jobs for *javaScript developer* are titled *frontend Engineer* or contain the keyword *jQuery* frequently. It means these keywords will be helpful for further recommendation [10].

Collaborative-filtering finds what the job-seeker applied to, Thereafter it tries to find the most similar jobs to this job x. It assumes if many applicants of who applied to job x applied also to job y, that means that job y is somehow similar to job x and will be interesting to the job-seeker who applied before to x. CF, BBR and COBR modules were also significantly helpful for generating a relational ontology for job-titles and keywords.

For example, detecting instances like ديلفري, ديليفيري, دلفري, دليفرى. They all are different used spelling variations for the same job *delivery boy* jobs in Arabic language. Another example is موظف شئون عاملين, شئون موظفين, مسئول موارد بشرية. They are detected as synonyms like *human resources officer*, *personnel specialist*, *hr*.

Knowledge-based recommendation solved the cold-start problem by retrieving explicitly preferred results and the tolerated results, the results that cannot retrieved through content-based recommendation. We analyzed our data to extract the generic rules of our users' behaviors. As example, we figured out that employers tolerate their jobs' age requirements but they are rarely tolerating regarding their jobs' gender requirements. It means that 22 years old job-seeker can be interested in another job which requires 24 minimum age and the employer will probably tolerate and accept this applicant.

Demographic-based is similar to knowledge-based but it extracts the specific rules of each segment behaviors (the case-study of education-based clustering is explained in the data section).

Ontology-based recommendation is a simple recommendation version which tries to fetch some other jobs which considered among the same hierarchy/specialty of the previous recommended jobs [11], like SEO, social media, e-marketing, market research are previously grouped and labeled as marketing jobs.

Algorithm. Generally, the algorithm of hybrid job recommendation flows as follow

```
For each user u
  Process KBR refined by DBR and BBR
  Run CBR refined by all of KBR, DBR and BBR
  Run CF by using crowd behavior and individual BBR
  If (recommendations are few)
    Generate COBR and OBR in conjunction with CBR
Return top ranked jobs J
```

3.2 Recommendation Search Queries (Textual Data)

The algorithm uses some keywords from each job-seeker's profile and previous behavior to search in our job index. As example, keywords and job titles of the jobs that job-seeker recently applied to, previous experiences, skills, previously conducted search queries. This information helps the recommender system to understand the preferences of each job-seeker. This combined data is manipulated to be passed to our search engine to retrieve textually matched jobs. We discussed the details of this textual similarity engine in another publication.

3.3 Recommendation Cutoff Filters (Attributes-Based Data)

It is not just about matching the textual content to produce meaningful recommendations. There is the attribute-based matching which is responsible for cutoff irrelevant jobs from recommendations. It manipulates the matching of the important attribute-based data that extracted from job-seeker and job profile (illustrated in the data section) like age, gender, educational level, salary…, etc. All these attributes are further manipulated also by our ranking model to rank the most relative jobs first as discussed in the ranking section.

4 Data

Data sample is collected from a real online robust e-recruitment platform. We divided the data into two parts. The first one is the content-based data, and the other one is the behavioral-based data (Fig. 2).

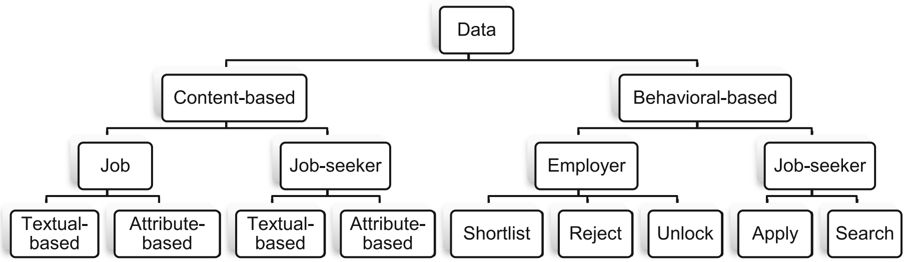


Fig. 2. Dataset hierarchy

Content-based data is filled by the user manually during the process of registering to the system. For job-seeker, these filled data could be attribute-based or textual-based. Attribute-based data can be gender, country, seniority, and educational level. But the textual-based fields allow users to fill them freely with any possible data, such as previous occupations, activities, skills, and trainings. On the other side, the job profile contains also attribute-based data such as required min/max age, gender, and offered salary. Textual-based fields of the job are job title, job description…, etc.

Behavioral-based data reflects the behavior of the user while interacting with the platform. Certainly every user type has different actions to perform. Job-seeker can search for specified jobs in addition to apply to the suitable job. The employer receives the applications of job-seekers, screens job-seekers' profiles to shortlist or reject to finally accept only who fit into that position and unlock the contact information of the job-seeker as a final step to be able of contacting him/her.

4.1 Feature Selection

Deciding which feature should be considered to be used in recommendation is another data mining task which should be done at the early stage. This process objective is to find the most influential features on the recommendation quality. The most influential features/attributes are defined based on four pillars: user surveys, job-seekers' online behavior, employers' online behavior and our domain-knowledge. As the deep discussion of these pillars is not the focus of this paper. Briefly this pure data mining task involves anomaly detections to ignore outliers, and association rules as support, confidence, lift and conviction to assert which job attributes will attract the job-seekers to apply, and which job-seeker attributes will bias the employer to shortlist.

Following is an example of analyzing job-seekers behavior with respect to job recency. The graph in Fig. 3 demonstrates the interest of job-seekers mainly is in recent jobs. Their applying trend reaches the peak on jobs that are posted only one day ago, but the applying demand starts in inclining. The applying demand for jobs older than 7 days forms 20% only of the total demand.

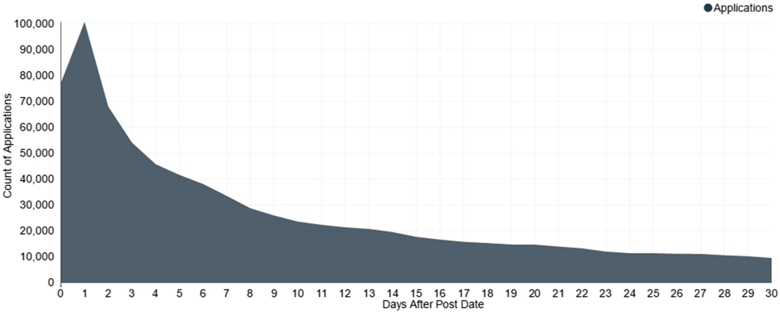


Fig. 3. Job applications frequency along time after job posting

By similar analysis to user surveys, job-seekers' online behavior and employers' online behavior and the usage of association rules of data mining beside to our domain knowledge, we generated Fig. 4 which is a specific cluster sample of education-based clustering for blue-collars job-seekers.

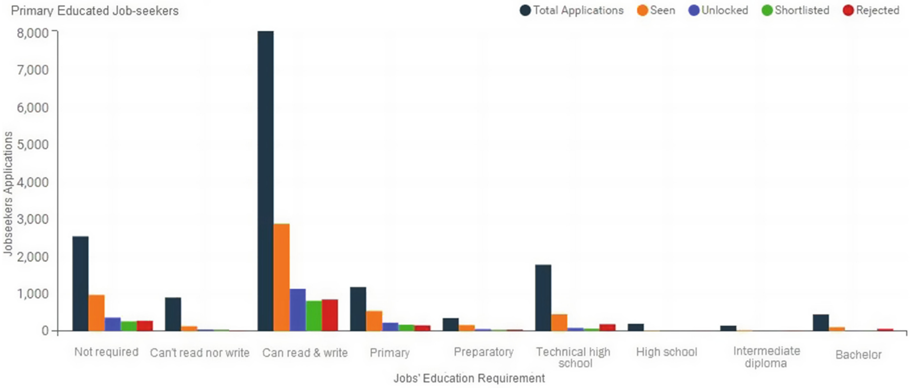


Fig. 4. Behavior of employers and job-seekers based on education clustering

Gray bar is the total job applications, orange bar is the seen applicants by the employer/recruiter, blue bar is the count of unlocked applicants by the employer/recruiter, green bar is the count of shortlisted applicants, red bar is the count rejected applicants and finally values on the y-axe are the different education levels required by the job.

Above chart shows clearly the interest of primary education job-seekers through the gray bar; on the other hand, it clarifies the employers' reactions through the other bars (orange, blue, green and red). It is easy to obtain that there is a high job-seekers interest in applying to jobs which require the following education: "not required", "can read & write" and "technical high school". Because of few jobs requiring primary education, the applications rate of primary education job-seekers on this segment is lower than others, therefore the count of jobs in each segment should be considered as well.

On the other side; employers who posted jobs with "technical high school" as a required education are not interested in those primary education job-seekers. But the other employers who posted "not required", "can read & write" and "primary" as education requirements are interested in this cluster of job-seekers, this observation is deducted through analyzing seen, shortlist and unlock activities.

Similar analyses are applied on all attributes to summarize the important attributes from job-seeker profile and job profile. Extracted attributes from job-seeker profile are gender, age, city, district, education, experiences' main work fields, experiences' specialties, preferred main work fields and preferred specialties. Extracted attributes from job profile are gender, minimum age, maximum age, city, district, minimum and maximum salary, education, job main work field, job specialty and job post date.

5 Ranking Algorithm

The ranking algorithm is trying to produce most related jobs in an ascending order by combining the relevancy, proximity and recency. Therefore, the first recommended job should be the newest and nearest most relevant job to each job-seeker.

5.1 Relevancy Ranking (RR)

This section discusses some of formulas that contribute to calculate the relevancy score. RR retrieves a set of recommended jobs J which is similar to the produced job-seeker profile P. That job-seeker profile P is generated by the aforementioned hybrid recommender engine. It means that the job-seeker profile contains the filled data (content), individual actions (behavior), crowd actions (collaborative)…etc. For example, job-seeker wrote *Angular.js* among his/her skills (content) and applied to *Frontend Developer* (behavior), and job-seekers who applied to *Frontend Developer* also applied to *UI Developer* (collaborative). All of these keywords *Angular.js, Frontend Developer* and *UI Developer* will form the job-seeker profile P to find the similar jobs. Jobs J that contains more of these terms will have higher relevancy score

$$J < P_1^{job\,title}, \ldots P_k^{job\,title}, P_1^{Skill}, \ldots P_y^{Skill}, \ldots P_1^{Keyword}, \ldots P_z^{Keyword} > \qquad (1)$$

For gender matching, jobs that require the same job-seeker's gender U^{gender} will have higher relevancy score than the non-matching jobs, the same concept is applied on other fields such as education. Another relevancy factor is the numerical attributes relevancy like age and salary. Age matching formula is defined as follow to retrieve only the jobs requiring the same age range of job-seeker's age U^{age}

$$J^{minAge} \leq U^{ge} \leq J^{maxAge} \tag{2}$$

However according to our conducted data analysis, there is some tolerance can be added to the age, so that the correct formula to recommend jobs regarding the age is

$$J^{minAge} - \delta \leq U^{age} \leq J^{maxAge} + \alpha \tag{3}$$

These two tolerance parameters δ and α are not constant. They are variables adjusted and increased along the recommendations generation. They are important influencer of the recommendation ranking formula. Therefore, the relevant jobs with lower values of δ and α are most likely to have a higher relevancy score than similar jobs with higher values of δ and α. For example the job that requires 25 minimum age ($\delta = 0$) will get a higher relevancy score than any job requires 26 minimum age ($\delta = 1$) when recommending job for a 25 year old job-seekers. The same concept is applied for salary matching. The most suitable job to job-seeker is the job that matches the exact desired job-seeker's salary. Actually job-seekers can accept job if the offered salary is a bit lower than the expected salary. But certainly job-seeker will not mind if the recommended job offers higher salary than the expected one.

$$J^{minSalary} - \varphi * U^{salary} \leq U^{expectedSalary} \leq J^{maxSalary} + \mu * U^{salary} \tag{4}$$

5.2 Proximity Ranking

For recommendations proximity, only jobs around job-seeker's residence $U^{residence}$ by variable radius β are retrieved.

$$J^{location} \leq U^{residence} + \beta \tag{5}$$

This variable radius is adjusted and expanded along the recommendations generation and used among an equation to define the proximity score of job. Figure 5 illustrates a case study based on Egypt map. It shows that Madinaty residents will get nearby jobs first before farther jobs, but they will not get jobs farther than the upper limit of β. According to this case study example, the upper limit of β is at 6th of October City. Location labeled with number 1 is Madinaty while location labeled with number 2 is 6th of October City and the red dotted line refers to β. This β is not a constant value. Actually it is a variable based on some parameters like the job-seeker ability to relocate & number of recommendations generated to this job-seeker. Recommended jobs with lower values of β have higher proximity score than other recommended jobs with β values.

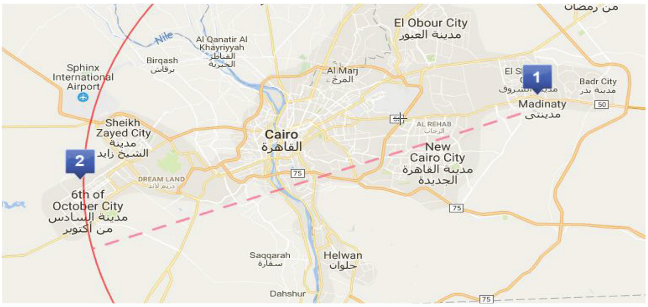


Fig. 5. Illustration of proximity recommendation

5.3 Recency Ranking

Job recency is another important factor that is extracted based on our data analysis that discussed previously. Therefore it is a vital factor to consider when thinking in results ranking. Jobs with older date have lower recency score than other recent jobs.

5.4 Hybrid Ranking Algorithm

Our proposed hybrid ranking algorithm (HRA) is composed of the aforementioned relevancy, proximity and recency ranking (Fig. 6). HRA computes all of relevancy, proximity and recency scores for each job. Then it uses these scores to return a sorted list of the recommended jobs ordered by relevancy, proximity and recency scores. It sorts jobs by their relevancy score. If a tie occurred (two jobs have an equal relevancy score), it sorts the tied jobs by proximity score. If another tie occurred (two jobs have an equal relevancy and proximity score), it sorts the tied jobs by recency score. The pseudo-code of HRA is illustrated as follow.

```
For each recommended job j to user u
  Compute a relevancy score j.rs between u and j
  Compute a proximity score j.ps between u and j
  Compute a recency score j.rcs between j and time.now()
  Insertion_sort(j,J) by inserting j into sorted J
Return top ranked jobs J, ordered by rs then ps then rcs
```

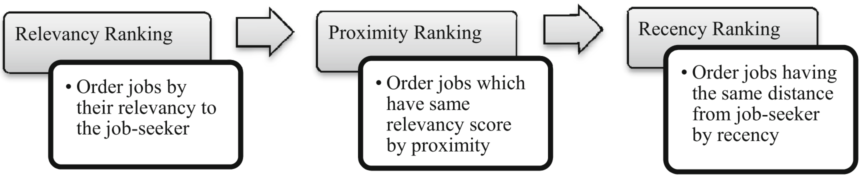

Relevancy Ranking
• Order jobs by their relevancy to the job-seeker
Proximity Ranking
• Order jobs which have same relevancy score by proximity
Recency Ranking
• Order jobs having the same distance from job-seeker by recency

Fig. 6. Recommended jobs ranking phases

6 Accuracy Evaluation

It is important when developing a new system to set some kind of criteria for evaluating that proposed system. There are numerous metrics whether offline or online metrics. Offline metrics are precision, recall, F1 measure and normalized discounted cumulative gain (nDCG), however the results of offline metrics may be misleading. Therefore online and real-world evaluations represent probably better methods to evaluate such platforms and assist precisely the users' satisfaction [12].

Online and real-world business metrics are user conversion rate, click-through rate (CTR), time to first click, first click rank. Job application conversion rate is one of the accuracy and business key metrics [12–14], it is about how many actions (apply) are conducted on recommended jobs relative to the total actions (apply) which we will adopt soon. But we evaluated via click rank, the result was that the first recommended job got the highest click rate then the second then the third job.

7 Conclusion

We presented our developed HIFE-JRS. The aim is developing an efficient online recruitment platform that gains the high level of satisfaction of both job-seekers and recruiters. This is achieved through designing a hybrid recommendation algorithm that is based on content, behavior, demographics…, etc. and by analyzing the most influential recommendation data. We also discussed how these attributes could be used as input to our hybrid algorithm in addition to the most efficient ranking criteria for producing the best job recommendation models.

References

1. Liu, J., Dolan, P., Pedersen, E.R.: Personalized news recommendation based on click behavior. In: Proceedings of the 15th International Conference on Intelligent User Interfaces, Hong Kong, China, pp. 31–40 (2009)
2. Zhu, C., Zhu, H., Xiong, H., Ding, P., Xie, F.: Recruitment market trend analysis with sequential latent variable models. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, California, USA, pp. 383–392 (2016)
3. Zheng, S.T., Hong, W.X., Zhang, N., Yang, F.: Job recommender systems: a survey. In: Proceedings of the 7th International Conference on Computer Science & Education (ICCSE 2012), Australia, pp. 920–924 (2012)
4. Lu, Y., Helou, S., Gillet, D.: A recommender system for job seeking and recruiting website. In: Proceedings of the 22nd International Conference on World Wide Web Companion, pp. 963–966 (2013)
5. Owen, S., Anil, R., Dunning, T., Friedman, E.: Mahout in Action. O'Reilly, Japan (2012)
6. Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.): Recommender Systems Handbook. Springer, US (2011). https://doi.org/10.1007/978-0-387-85820-3
7. Al-Otaibi, S.T., Ykhlef, M.: A survey of job recommender systems. Int. J. Phys. Sci. **7**(29), 5127–5142 (2012)

8. Pandya, S., Shah, J., Joshi, N., Ghayvat, H., Mukhopadhyay, C., Yap, M.H.: A novel hybrid based recommendation system based on clustering and association mining. In: Proceeding of the 10th International Conference on Sensing Technology, China (2016)
9. Hong, W., Zheng, S., Wang, H., Shi, J.: A job recommender system based on user clustering. J. Comput. **8**(8), 1960–1967 (2013)
10. AlJadda, K., Korayem, M., Ortiz, C., Russell, C., Bernal, D., Payson, L., Brown, S., Grainger, T.: Augmenting recommendation systems using a model of semantically-related terms extracted from user behavior. In: Proceeding of the Second CrowdRec Workshop RecSys, Austria, pp. 1409–1417. ACM (2014)
11. AlJadda, K., Korayem, M., Grainger, T., Russell, C.: Crowdsourced query augmentation through semantic discovery of domain-specific jargon. In: Proceeding of IEEE International Conference on Big Data (Big Data), USA (2014)
12. Beel, J., Langer, S.: A comparison of offline evaluations, online evaluations, and user studies in the context of research-paper recommender systems. In: Proceeding of the 19th International Conference on Theory and Practice of Digital Libraries, Poland (2015)
13. Liu, R., Ouyang, Y., Rong, W., Song, X., Tang, C., Xiong, Z.: Rating prediction based job recommendation service for college students. In: Proceeding of International Conference on Computational Science and its Applications (ICCSA), China (2016)
14. Paraschakis, D., Nilsson, B.J., Hollande, J.: Comparative evaluation of Top-N recommenders in e-commerce: an industrial perspective. In: Proceeding of IEEE 14th International Conference on Machine Learning and Applications, USA (2015)

Behaviorally-Based Textual Similarity Engine for Matching Job-Seekers with Jobs

Islam A. Heggo(✉) and Nashwa Abdelbaki

School of Communication and Information Technology, Nile University, Giza, Egypt
{i.heggo,nabdelbaki}@nu.edu.eg

Abstract. Understanding both of job-seekers and employers behavior in addition to analyzing the text of job-seekers and job profiles are two important missions for the e-recruitment industry. They are important tasks for matching job-seekers with jobs to find the top relevant suggestions for each job-seeker. Recommender systems, information retrieval and text mining are originally targeted to assist users and provide them with useful information, which makes human-computer interaction plays a fundamental role in the users' acceptance of the produced suggestions. We introduce our intelligent framework to help build the knowledge required to produce the most relevant jobs based on processing each job-seeker profile's text, the behaviorally collected text and the jobs' profile content. We analyzed the available textual similarity scoring algorithms to find the best suitable relevancy ranking model which is plugged into our developed textual similarity engine. The main purpose is enhancing the recommendation quality in the challenging domain of e-recruitment by finding the textually similar jobs for each job-seeker profile.

Keywords: Information retrieval · NLP · RS · Relevancy scoring · Search engine Ranking · Personalization · TF-IDF · BM-25 · Textual matching

1 Introduction

Information retrieval, text mining, natural language processing and information filtering are the concepts that can be utilized to find material (jobs in our case) of an unstructured corpus (jobs' descriptions and requirements) that satisfies our user's needs within a large amount of data. The profiles of job-seekers and jobs have some structured data like the information that has predefined finite set (e.g. gender, educational degree), they also have some unstructured data like the information that does not belong to a definite set of options (e.g. job description, job-seeker previous responsibilities). That needs analyzing all this textual data to recommend only the matching and relevant jobs. But text analysis is not the only important mission to achieve our goal. Our ultimate goal is to match job-seeker with job, This requires the knowledge of recommender systems field as well. The idea is to find for users a personalized content that is suitable for each user interests. It predicts the interest that a user would give to an item. Recommender system can be behavioral-based or content-based. Behavioral-based is the concept of

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 564–574, 2018.
https://doi.org/10.1007/978-3-319-74690-6_55

tracking users' activities to understand the implicit interest of each user [1]. Content-based is the concept of considering the manually pre-filled information to understand the explicit interest of each user. Recommender systems become too popular in many applications such as movies, books, news, feeds, friends, courses and hotels recommendation. This paper discusses the various stages of a textual-based recommender engine by matching the content of job-seekers and jobs. It begins with the various conceptual information retrieval and filtering models to return the matching jobs only, text processing and matching layers to extract the significant terms for matching and the different relevancy scoring formulas for ranking results based on their textual relevancy score.

2 Literature Survey

Despite the significance of large-scale textual similarity engines on this vital domain of e-recruitment, very little academic research has been conducted on the details of these engines. It is rare to find researches discussing a methodology of analyzing job and job-seekers profiles or job relevancy ranking approach. However, we utilized the concepts of text mining, information retrieval and NLP from similar industries to address the topic of building an efficient similarity engine that can exploit the influential information present in job-seeker profiles and job posts. We investigated the leading job search engines LinkedIn and Glassdoor. They use users' historical actions such as search, view job, and apply to job, in addition to the explicit filled preferences. These data are manipulated to fetch similar job posts. LinkedIn and Glassdoor rely mainly on keyword-based search to query their pool of jobs. Despite they are great systems, sometimes they recommend irrelevant jobs that could be biased to user's behavior rather than user's profile content (e.g. skills, experiences). For instance, users' profiles contain experiences such as senior software engineer, lead software engineer and product manager. Nevertheless, LinkedIn keeps recommending jobs like database administrator and project coordinator.

3 Hybrid Textual Matching Methodology

We propose the methodology of analyzing both of job-seekers profiles and job posts. Our methodology starts by deciding the influential type of data for an efficient recommendation before discussing the needed information retrieval, processing and ranking models.

3.1 Behavioral-Based and Content-Based Extracted Text

It is important to collect more text from various sources to produce accurate results. It is the first phase of deciding the text that should be considered for matching. From the job side, it is obvious that considering most of the text in the job post is useful. On the job-seeker side, not only the job-seeker's profile content but also considering the behavioral-based information will lead to higher matching and recommendation quality. There is valuable information can be collected from the frequent behavior of users [2–4].

For example, tracking the jobs which the user had applied to, include only the most frequent job-titles and keywords from these jobs and use them in the upcoming recommendation queries. The most frequent keywords included in the job-seeker search queries and filters are also ultimately useful to understand what the job-seeker is looking for when these queries are frequently and periodically conducted. Getting the continuously viewed jobs by job-seeker and how long each visit takes is another good factor to consider, but it is less important than the searched queries and previous applications. Therefore using some behaviorally collected keywords will be definitely useful Fig. 1.

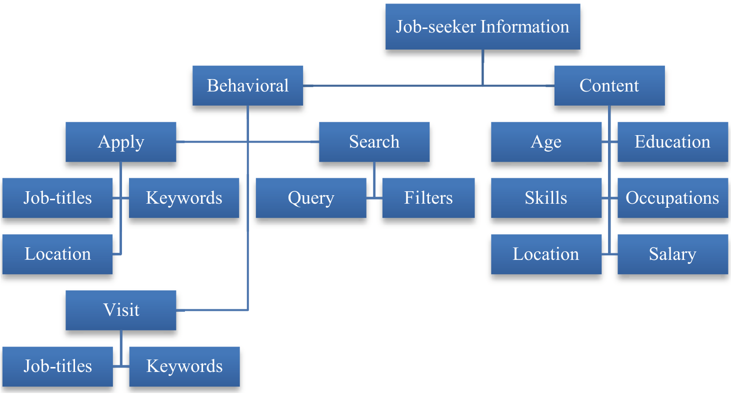


Fig. 1. Behavioral-based and content-based collected text

3.2 Boolean Independence Model of Information Retrieval (BIM)

It is a very simple model based on sets theory and boolean alegbra. It considers documents as a set of terms and the queries are expressions over terms. Query could be a simple query which is one term or multi-terms which contains multiple of terms. If the document matches the query expression then it is approved as a relevant document [5, 6]. As shown in Table 1, it contains record for each document whether it contains each term out of all the different terms or not. Each cell has 1 if the corresponding term is found in the corresponding doc and 0 if term is not found.

Table 1. Representation of presence and absence of terms in documents

	Doc #1	Doc #2	Doc #3	Doc #4	Doc #5	Doc #6	Doc #7
Software	1	0	1	0	1	0	0
PHP	**0**	0	0	**1**	0	1	0
Solr	**1**	1	1	**0**	1	0	1
Search	**1**	1	0	**1**	1	0	1
Engine	**0**	0	1	**1**	0	1	0

- Consider the following query

  (Search AND Engine) OR PHP OR Solr

- The corresponding boolean algebra expression is as follow

  (Search ∧ Engine) ∨ PHP ∨ Solr

- That expression will return two documents:- Doc #1, Doc #4 (Table 2).

Table 2. The corresponding applied boolean algebra to BIM example

<table>
<tr><th></th><th colspan="4">(Search ∧ Engine) ∨ PHP ∨ Solr</th><th></th><th colspan="4">(Search ∧ Engine) ∨ PHP ∨ Solr</th></tr>
<tr><td rowspan="3">Doc #1</td><td>1</td><td>0</td><td>0</td><td>1</td><td rowspan="3">Doc #4</td><td>1</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td colspan="2">0</td><td colspan="2">1</td><td colspan="2">1</td><td colspan="2">1</td></tr>
<tr><td colspan="4"><b>1</b></td><td colspan="4"><b>1</b></td></tr>
</table>

3.3 Vector Space Model of Information Retrieval (VSM)

A vector space model or term vector model is a model where two vectors are defined. First one is the query vector and the second vector is for the document. Each term in the query is considered as vector dimension in the query vector. The relevance of the document to the query is obtained through calculating the scalar product of those two vectors [5, 6]. It manipulates document like a bag of words. A document is a t-dimensional vector $d = [d_1, d_2 \ldots, d_t]$. The i^{th} entry of the vector is the boosting weight or importance of term d_i in the document. A query is a t-dimensional vector as well $q = [q_1, q_2 \ldots, q_t]$. The i^{th} entry of the vector is the boosting weight or importance of term q_i in the query. Then the dot product of above defined vectors is calculated (Eq. 1). Each document is scored according to its matching to the query. The dot product produces a non-negative real number. The document matches the query if its score is greater than zero. Only documents with non-zero score are returned

$$q.d = \sum\nolimits_1^t q_i d_i = q_1 d_1 + q_2 d_2 + \ldots + q_t d_t \tag{1}$$

Basically we can set each dimension in the document vector to one if the document includes the term d_i and zero if the document does not include it. Instead, there are more accurate boosting weights can be set, like the term frequency in the document to differentiate between documents that include the term once and other documents which use the term more frequently. So if a document includes specific term 10 times then it will have 10 as a term weight (d_i) in the document vector.

3.4 Textual Processing and Matching Layers

We need to prepare the text and emphasize on the crucial factors that can significantly produce better recommendations. We adopted many concepts to enhance the textual relevancy matching between job post and job-seeker profile.

Term Identification. Number of matched query terms is proportionally correlated to the document relevancy. In some cases the query terms can be ORed terms not ANDed terms, then the most relative documents are the ones which have more query terms.

Typo and Misspelling Tolerance (Fuzzy Search). It depends on Damerau–Levenshtein distance algorithm. It is the advanced algorithm of Levenshtein distance. Levenshtein algorithm is built to measure the similarity between two strings by counting minimum number of single-character operations required to transform one word into the other. These operations include character deletions, character insertions and character substitutions. Damerau–Levenshtein added the transposition of two adjacent characters operation. It is important for retrieving relevant results even if the query was mistyped. For example, if the query is *Jave developer*. Damerau–Levenshtein will figure out that this query needs one character substitution to match *Java developer* by substituting the "e" to "a" in *Java*.

Terms Proximity. If the query has two terms or more, proximity role is about identifying how physically near are those terms in the relevant matched documents. Example if query is software engineer, then the engine will retrieve documents which have software engineer where its proximity can be considered as zero because it is the same like search query, but it is reasonable to retrieve documents which contains software development engineer where its proximity can be considered as one because there is one word "development" which is laid between the query terms "software" and "engineer".

Attribute Importance and Field Weights. The attributes importance criterion identifies the most important matching attribute(s) of the document. Basically each stored document has many attributes as job titles, job description… etc. So if user searches for "operation manager", it is meaningful to get jobs have job titles similar to "operation manager" in the top list, preceding other jobs that have "operation manager" in the job description.

Tokenization. It is the process of splitting corpus based on our defined rules and producing single tokens and terms to achieve an efficient textual matching between job-seeker and job profiles [7]. Our rules are based on tokenizing on whitespaces, character-numeric transition and some special characters. An example of character-numeric transition is "PHP5" and "CSS3" will be tokenized to "PHP" "5" and "CSS" "3", these tokens will match any other profile contains "PHP5", "CSS3", "PHP" and "CSS" without versioning. An example of special characters tokenization is "asp.net" and will be tokenized to (asp) (net), these tokens will match any other profile contains "asp.net" "asp" and ".net"

Term Frequency (TF). It means that the most relevant document which has the highest term frequency, term frequency refers to the number of times that term appeared in the document [5–7]. Example like searching for *accountant*, the first retrieved document should be the one which contains the maximum count of word *accountant*.

Term Novelty, Inverse Document Frequency (IDF). It tends to obtain the term's importance. It needs to define a weight for each term according to their importance, so if a term is so common in all documents then it should get a low score than the other terms which more meaningful [5–7]. Example like searching for "HR at Vodafone UK", the term "UK" here is meaningless if all jobs is already in UK, it will get bad results if the engine emphasized on that term to return the documents which could contain dozens of UK, instead it should return the documents which contain highest occurrences of "HR" and "Vodafone".

Text Normalization. It is used to transform all characters variations to only one variation to simply retrieve the same term in case of writing it in different variation [5–7]. This layer is important for some languages like Arabic. In Arabic, we need to normalize all forms of "Alef" to only one form (إ , أ , ا , آ), the same is applied for all "teh" (ة, ه) and dotless "yeh" (ي, ى). For Arabic, removing any type of diacritics and character stretching is required. Example of diacritics is adding "damma", "shadda", "fatha" or "kasra" like (مُمَثِل). An example of stretching characters in Arabic as (تطويـــــــــــــل).

Stop Words. For better reasonable search results, it is important to remove the most common useless words that contain no valuable meaning in our context such as the, in, at, it, for, on, is, …, etc.

Synonyms. Using synonyms is the approach of query augmentation by including the similar words or synonyms [4, 8]. It is helpful to append the similar words to achieve wide range of accurate results. This tells our matching engine to behave similarly with the similar words as *HR*, *human resources* and *personnel specialist* or *senior* and *Sr.* There are many synonym sets, but it is dangerous to use a generic synonym sets. [9] The ultimate solution is to build your own domain-based set to be accurate. For example, in Arabic شيخ could be a synonym for "عجوز" but "شرم الشيخ"" is a city that cannot be a synonym for شرم العجوز". Another English example is *substitute* which is a synonym for *alternative*, but we cannot augment *jquery-substitute* with *jquery-alternative*. The context is very important. Polysemy problems which means that a word can have more than one meaning depending on the context like *fair* could mean *exhibition* or the adjective of *fairness* which means *unbiased*, another Arabic example is "طيار" could mean *pilot* or *delivery boy* in some Arabic slang languages.

Capitalization and Case Folding. It is often convenient to normalize and lower case every character for better matching. That will match 'MySQL', 'mysql' and 'Mysql'. Counterexamples include 'IT' vs. 'it' and 'US' vs. 'us'. So it is influential to handle that carefully [10].

HTML/Tags Cleansing. Removing unwanted tags and characters is another important cleaning layer [10]. Such layer will empower our engine to eliminate tags such as <u> html tags to avoid retrieving irrelevant results when searching for *UI* which refers to *User Interface*.

Stemming. It is the process of transforming the term into its origin form. Aggressive English stemmer (Snowball, Porter) will stem *development*, *developers* into *develop.* Counterexamples, it stems *international* to *intern*, *engineering, engine* to *engin* and *accountant* to account. K-stemmer, is a light English stemmer that stems *developers* to *developer*. There is also a fine light Arabic stemmer that is developed by Larkey et al. [11]. It removes many affixes (e.g. وبالامارات to امارات).

3.5 Relevancy Ranking Models

There are many developed ranking formulas to score the relevant documents. There are Okapi BM25 and different customizations of TF-IDF which are widely involved in search engines. They are ranking functions to rank matching documents according to their relevancy to a given search query. We utilized these equations to score each job and rank them based on their matching level to the job-seeker profile [12–14].

Naive TF-IDF. It is the simplest variation of TF-IDF which is defined as the multiplication of term frequency by inverse document frequency [13].

$$tf - idf = tf(t, d) \,.\, idf(t) \tag{2}$$

Where tf(t, d) is the raw frequency of a term in a document, it is the number of times that term *t* occurs in document *d*.

$$tf(t, d) = frequency \tag{3}$$

idf(t, d) is the inverse document frequency, which is the logarithm of dividing total number of documents (*n)* by the number of documents (n_t) contain the term *t*

$$idf(t) = \log\left(\frac{n}{n_t}\right) \tag{4}$$

Adjusted TF-IDF Scoring Function. More sophisticated TF-IDF is clarified below

$$score(q, d) = coord(q, d) \cdot queryNorm(q) \cdot \sum\nolimits_{t \in q} \left(tf(t, d) \cdot idf(t)^2 \cdot t.getBoost() \cdot norm(t, d)\right) \tag{5}$$

coord(q, d) is a score factor based on how many of the query *q* terms are found in the document *d*. The more query terms in a document the higher is the score [14].

queryNorm(q) is just a factor for normalizing, it is used to make scores between queries comparable, but it does not affect individual document score as it is applied to all documents. All ranked documents are multiplied by the same factor

$$queryNorm(q) = \frac{1}{\sqrt{q.getBoost()^2 + \sum_{t \in q} (idf(t) \cdot t.getBoost())^2}} \tag{6}$$

t.getBoost() is a search time boost of term *t* in the query *q*. *tf(t, d)* stands for term *frequency*, it is the number of times the term *t* appears in the currently scored document *d*. The more term occurrences in a document the higher is the score

$$tf(t, d) = \sqrt{frequency} \tag{7}$$

idf(t) correlates inversely to n_t (the number of documents where the term *t* appears). *n* is the number of all stored documents. This means rarer terms will highly influence the total score. This IDF formula is modified in a minor way to handle a term appearing in no documents (hence avoiding a zero denominator)

$$idf(t) = 1 + \log\left(\frac{n}{n_t + 1}\right) \tag{8}$$

norm(t, d) is another normalization function. It is responsible of document boosting, field boosting and document length normalization. *lengthNorm* is computed in accordance with the number of tokens or terms of this field in the document, so shorter fields have higher influence on the score.

$$norm(t, d) = doc.getBoost() \cdot lengthNorm \cdot \prod_{f \in d} f.getBoost() \tag{9}$$

BM25 Scoring Function. Okapi BM25 is a probabilistic retrieval and ranking model.

$$score(q, d) = \sum_{t \in q} \left(idf(t) \cdot \frac{tf(t, d).\left(k_1 + 1\right)}{tf(t, d) + k_1.\left(1 - b + b.\frac{|d|}{avgdl}\right)} \right) \tag{10}$$

k_1 is a variable to control non-linear term frequency normalization (saturation), default value is 1.2. In classic Lucene, TF is constantly increases and never reaches a saturation point, so this modification is adjusted to suppress the impact of term frequency. |*d*| is the length of the current matching document and *avgdl* is the average stored documents length. These last two variables compute how long a document is relative to the average document length, so relatively longer document than the average length will get lower score than shorter documents. b is another variable to control to what degree document length normalizes TF values, by default it equals to 0.75, it is important to finely tune how much document length influence the scoring function. The inverse document frequency of BM-25 is defined as follow.

$$idf(t) = \log \frac{n - n_t + 0.5}{n_t + 0.5} \tag{11}$$

TF-IDF vs BM-25. TF-IDF is a vector space model but BM-25 is defined as one of probabilistic models. However their concepts are not very different. They both consider some kind of weights for each term as the result of calculating the product of some IDF formula variation and some TF formula variation to produce that term weight as a relevancy score of the current processed document to the given query. However, one of the most important differences between TF-IDF and BM-25 is the saturation when term appears more frequently. They both agree on giving high score for documents contain higher terms occurrences. But the impact of term frequency is always rising, therefore BM25 typically approaches a boundary for high term frequencies. Other classic TF constantly increases and does not reach a reasonable boundary. With exclusion of the document length, BM-25 term frequency formula is defined in Eq. 12. The variable k is often equal to 1.2. It is a variable that can be configured from 1.2 to 2.

$$\frac{tf(t,d).(k+1)}{tf(t,d)+k} \tag{12}$$

Adjusting k is useful to alter the influence of TF, it controls the saturation boundary. Higher k values leads to further saturation reach. Through stretching out the point of saturation, it stretches out the relevance score difference between documents with higher TF and other documents with lower TF, as shown in Fig. 2.

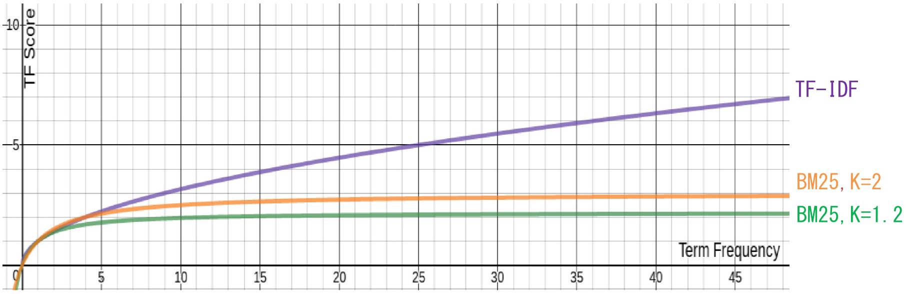


Fig. 2. Term frequency saturation of TF-IDF and BM-25

4 Results and Deductions

This textual similarity engine is developed as a part of hybrid job recommender system. There are many criteria for evaluating recommender systems whether offline metrics such as recall, precision, F1 measure and normalized discounted cumulative gain (nDCG) or online real-world metrics such as click-through rate (CTR), user conversion rate, time to first click and first click rank. Our data sample is collected from a real online robust e-recruitment platform. It contains 150,000 jobseekers, in addition to 10,000 unique jobs and 650,000 job applications. We initially evaluated the whole hybrid recommender system via click rank metric. The result was that the first recommended job got the highest click rate then the second job then the third one. Our experimental results to this standalone textual matching layer and ranking technique lead to the following desired deductions: - jobs matching more query terms of the job-seeker's keywords are getting higher score than jobs matching less query terms. Jobs having more occurrences of a query term (job-seeker's keywords) are ranked higher than jobs with fewer occurrences of terms. Jobs containing more novel terms have higher score than jobs containing common terms, rare terms have more significance than the common terms. Because of using the documents' length normalization, shorter job posts having the same occurrences of query terms are getting higher score than lengthy job posts, it means this job concentrates more on these terms (skills). Jobs including query terms in the job title have better score than other jobs including terms in less significant fields such as job description (terms existence in more important fields is more significant).

5 Conclusion

The paper presents a textual similarity engine for matching the job-seekers profiles with the relevant jobs. The aim is building a powerful e-recruitment platform that serves efficiently both of job-seekers and employers to gain their highest level of satisfaction. We worked on reaching this by illustrating the methods of matching search query with a set of documents and how these methods could be utilized by using each job-seekers' keywords as our search query and handling jobs as our retrievable documents. We defined the needed algorithms and layers to process the text to retrieve the most matching jobs such as tokenizing profiles, eliminating stop words, normalizing and removing diacritics from text, using the frequent and novel tokens…etc. We also analyzed the different relevancy scoring formulas to rank the jobs based on their relevancy to the job-seeker profile content. We focused particularly on comparing TF-IDF and BM-25 formulas for better understanding of their advantages and to be utilized correctly based on different case-studies. Finally we showed the influence of our developed model and how it is really affecting the results ranking and qualified to produce the best matching quality for job-seekers.

References

1. Liu, J., Dolan, P., Pedersen, E.R.: Personalized news recommendation based on click behavior. In: Proceedings of the 15th International Conference on Intelligent User Interfaces, Hong Kong, China, pp. 31–40 (2009)
2. Lu, Y., El Helou, S., Gillet, D.: A recommender system for job seeking and recruiting website. In: Proceedings of the 22nd International Conference on World Wide Web Companion, Rio de Janeiro, Brazil, pp. 963–966 (2013)
3. Rafter, R., Bradley, K., Smyth, B.: Automated collaborative filtering applications for online recruitment services. In: Brusilovsky, P., Stock, O., Strapparava, C. (eds.) AH 2000. LNCS, vol. 1892, pp. 363–368. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-44595-1_48
4. AlJadda, K., Korayem, M., Ortiz, C., Russell, C., Bernal, D., Payson, L., Brown, S., Grainger, T.: Augmenting recommendation systems using a model of semantically-related terms extracted from user behavior. In: Proceedings of the Second CrowdRec Workshop. ACM RecSys, Austria (2014)

5. Büttcher, S., Clarke, C., Cormack, G.V.: Information Retrieval: Implementing and Evaluating Search Engines. MIT Press, Cambridge (2016)
6. Krishnamurthy, S., Akila, V.: Chapter 2: Information retrieval models: trends and techniques. In: Web Semantics for Textual and Visual Information Retrieval, pp. 17–42 (2017)
7. Smiley, D., Pugh, E., Parisa, K., Mitchell, M.: Apache Solr Enterprise Search Server - Third Edition (2015)
8. AlJadda, K., Korayem, M., Grainger, T., Russell, C.: Crowdsourced query augmentation through semantic discovery of domain-specific jargon. In: Proceeding of IEEE International Conference on Big Data (Big Data), USA (2014)
9. Miller, G.A.: WordNet: A lexical database for English. Commun. ACM **38**(11), 39–41 (1995)
10. Jayalakshmi, T., Chethana, C.: A semantic search engine for indexing and retrieval of relevant text documents. In: The International Journal of Advance Research in Computer Science and Management Studies (IJARCSMS), vol. 4 (2016)
11. Larkey, L.S., Ballesteros, L., Connell, M.E.: Light stemming for arabic information retrieval. In: Soudi, A., Bosch, A., Neumann, G. (eds.) Arabic Computational Morphology. Text, Speech and Language Technology, vol. 38, pp. 221–243. Springer, Heidelberg (2005)
12. Ramos, J.: Using TF-IDF to Determine Word Relevance in Document Queries (2003)
13. Nayrolles, M.: Chapter 4: Relevancy and scoring mechanisms. In: Mastering Apache Solr: A Practical Guide to Get to Grips with Apache Solr, pp. 60–62 (2014)
14. Shahi, D.: Chapter 8: Solr scoring. In: Apache Solr: A Practical Approach to Enterprise Search, pp. 189–207 (2015)

Adaptive Task Scheduling on Multicore Processors

Samar Nour, Shahira Mahmoud(✉), and Mohamed Saleh(✉)

Helwan University, Cairo, Egypt
samar.nour@bue.edu.eg, shahira_heikal@h-eng.helwan.edu.eg,
mohamed.saleh@hq.helwan.edu.eg

Abstract. Nowadays multicores design become more complex, where they integrate different components on a single chip. Multithreading aims to increase utilization of a single core and decrease execution time. This research paper proposed adaptive scheduling in Multi2Sim framework to improve the performance of multicore-multithreaded processors. The performance evaluation of the adaptive scheduling shows minimizing execution time and maximize utilization compared types of scheduling in Multi2Sim framework.

Keywords: Multi2Sim framework
Multicore-multithreaded processors · Adaptive scheduling
Utilization · ILP · TLP

1 Introduction

Current multi-cores are supported multi-thread. Multithreading is similar to multitasking but enables the processing of multiple threads at one time, rather than multiple processes. Since threads are more basic instructions than processes, multithreading may occur within processes and is a way to increase performance through parallel processing. It is the ability of the microprocessor to process multiple hardware threads of execution as well as process multiple software threads.

The current generation of superscalar multicore is the result of many efforts of designing deep and wide pipelines that highly exploit instruction level parallelism (ILP). However, the potential of ILP present in current workloads is not high enough to continue increasing hardware units utilization. On the other hand, thread level parallelism (TLP) enables to exploit additional sources of independent instructions to maintain processor resources with a higher occupation. This idea, jointly with an overcome of hardware constraints, resulted in CMPs (chip multiprocessors), which include various cores in a single chip [1]. Each core can integrate either a simple in-order multithreaded pipeline [2] or a more complex out-of-order pipeline [3].

Task scheduling can be divided into static, dynamic and adaptive scheduling. In static scheduling, the assignment of tasks to processors is done before program execution begins. On the other hand, dynamic scheduling is based on

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 575–584, 2018.
https://doi.org/10.1007/978-3-319-74690-6_56

the redistribution of processes among the processors during execution [4]. An adaptive scheduler is the one which takes many parameters into consideration in making its decisions [5]. Our proposed is concerned with the efficient utilization of all the cores and resources for this purpose we need an efficient task scheduling technique. The research on computer architecture simulator is very important because simulator serves as an important tool for developing computer system architectures and software. Parallel simulation has been an active research topic for several decades. Several parallel simulation techniques have been proposed to address the performance issue [16].

In our work, we used Multi2Sim [17] as a simulation framework for CPU-GPU (Central processing unit - Graphics processing unit) heterogeneous computing. It includes models for superscalar, multithreaded, and multi-core CPUs, as well as GPU architectures. Multi2Sim is an open-source simulator that used C programming language. It can be downloaded from web site [18]. A lot of modifications were performed in Multi2sim. The result shows improvement in performance comparing to the raw one [7].

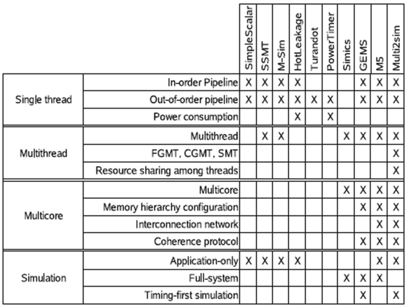
Our study aiming to characterize the true performance possibility of an adaptive scheduling. We design an optimal schedule that improves the performance of Mulit2Sim Framework Simulation.

The rest of the paper is organized as follows. Section 2 presents related work. In Sect. 3 Parallel Architectures in Multi2Sim is presented, Sect. 4 introduces an optimal scheduler using adaptive scheduling. In Sects. 5 and 6, the results of the quantitative evaluation review and finally conclusions are introduced.

2 Related Work

The evolution of multicore, mainly enabled by technology advances, has led to complex designs that combine multiple physical processing units on a single chip. These designs provide for the operating system the view of having multiple processors, and thus, different software processes can be scheduled at the same time. This processor model consists of three major components: the microprocessor cores, the cache hierarchy, and the interconnection network. A design modification on any of these components can affect the rest of them and cause-specific global behaviors. Therefore, the entire system should be modeled in a single tool that tracks the interaction between components. An important part of the work of this paper has focused on the development of the Multi2Sim simulation framework, which covers the limitations of other existing multiprocessor simulators. Multi2Sim integrates a model of the processor cores, the memory hierarchy, and the interconnection networks in a tool that enables their joint evaluation.

Multiple simulation environments aimed to evaluate computer architecture proposals which have been developed. The most widely used simulator in recent years has been SimpleScalar [8], which framework an out-of-order superscalar processor. A lot of extensions has been applied on top of SimpleScalar to model in a more accurate manner certain aspects of superscalar processors. For example, the HotLeakage simulator quantifies leakage energy consumption. SimpleScalar is quite difficult to extend for modeling new parallel microarchitectures without significantly changing its structure. In spite of this, various SimpleScalar extensions to support multithreading have been implemented. For example SSMT, M-Sim, or SMTSim, but they have the limitation of only executing a set of sequential workloads and implementing a fixed resource sharing strategy among threads. Multithread and multicore extensions have been also applied on top of the Turandot simulator [9,10], which models a PowerPC architecture. This tool has also been used with power measurement aims in an implementation called PowerTimer [11].



		SimpleScalar	SSMT	M-Sim	HotLeakage	Turandot	PowerTimer	Simics	GEMS	M5	Multi2sim
Single thread	In-order Pipeline	X	X	X	X				X	X	X
	Out-of-order pipeline	X	X	X	X	X	X		X	X	X
	Power consumption				X		X				
Multithread	Multithread		X	X				X	X	X	X
	FGMT, CGMT, SMT										X
	Resource sharing among threads										X
Multicore	Multicore							X	X	X	X
	Memory hierarchy configuration								X	X	X
	Interconnection network									X	X
	Coherence protocol								X	X	X
Simulation	Application-only	X	X	X	X					X	X
	Full-system							X	X	X	
	Timing-first simulation								X		X

Fig. 1. Main features of existing simulators.

In contrast to the application-only simulators, a set of so-called full-system simulators are available. Similarly to virtual machines, these tools boot the unmodified operating system. Although this model provides higher simulation power, it involves a huge computational load and unnecessary simulation accuracy depending on the goal of the study. Simics [12] is an example of a generic full-system simulator, commonly used for multiprocessor systems simulation, but unfortunately, it is not freely available. A variety of Simic's derived tools have been implemented for specific research purposes in this area. This is the case of GEMS [13], which introduces a timing simulation module to model a complete processor pipeline, a memory hierarchy, and cache coherence. However, GEMS provides low flexibility to model multithreaded designs and does not integrate an interconnection network model, while still adding a sensible amount of computational overhead and sometimes prohibitive simulation times. An important feature included in some processor simulators is the timing-first approach, provided by GEMS and adopted by Multi2Sim. In such a scheme, a timing module traces the state of the processor pipeline while instructions traverse it, possibly in a speculative state. Then, a functional module is called to actually execute the instructions, so the correct execution paths are always guaranteed by a previously developed the robust functional simulator. The timing-first approach confers efficiency, robustness, and the possibility of performing simulations at different levels of detail. Multi2Sim adopts the timing-first simulation with a functional support that, unlike GEMS, need not simulate a whole operating system, but is still capable of executing parallel workloads, with dynamic threads creation. The last cited simulator is M5 [14], which provide support for out-of-order SMT-capable CPUs, multiprocessors and cache coherence, and runs in both full-system and application-only modes. The limitations lie in the low flexibility of multithreaded pipeline designs. Figure 1 gathers the main simulator features and marks the differences between them [15]. The next sections focus on the design and implementation of some components of the baseline Multi2Sim tool, and it is discussed how the architectural techniques proposed in this paper are modeled on top of it.

3 Multi2Sim Parallel Architecture

Multi2Sim can supported parallel architectures, where the processor contains n cores, m nodes and one pipeline, as shown in Fig. 2.

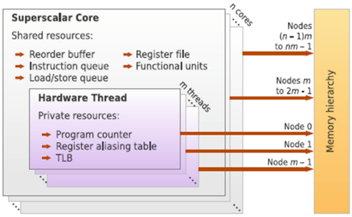


Fig. 2. Parallel architecture scheme.

A core is formed of one or more threads. It does not share any pipeline structure, execution resource, or queue with other cores, and the only communication and contention point among cores is the memory hierarchy.

A node is the minimum hardware entity required to store and run one task. In a multithreaded processor, each thread is one node. Finally, an n-core, t-threaded processor (meaning that each core has t threads) they have n * t nodes since they can store and run n * t tasks simultaneously. Each node can have its own entry point to the memory hierarchy to fetch instructions or read/write data. The number of processing nodes limits the maximum number of tasks that can be executed at a time in Multi2Sim.

Based on this definitions, Fig. 2 represents the structure of the parallel architecture modeled in Multi2Sim. Specifically, the figure plots a processor with 2 cores and 2 threads, forming 4 processing nodes with independent entry points to the memory hierarchy.

4 The Scheduler in Multi2Sim

Multi2Sim presents the concept of the task scheduling, similar to the idea of process scheduling in an operating system. The schedule is aimed to map software contexts to nodes (hardware threads) and run them. There are two types of task scheduling (Static and Dynamic) [18]. We aim to improve Multi2Sim performance, by designing adaptive scheduler which gives the best result compared to the previous schedules, sees the result section.

4.1 The Static Scheduler

The processing element selection and the start times are decided at compile time. The advantage of this approach is that the scheduling overhead is low. On the other hand, the Static Scheduler is very rigid and can not take advantage of the runtime state of the system. These properties are considered to be the static scheduler main disadvantages.

The static schedule is implemented in Multi2Sim. This type of scheduling maps tasks to hardware threads in a definitive manner, using the following criterion:

1. The followed allocation order maps first threads within a single core and then goes to the next core after the first one fills up.
2. This allocation is definitive, meaning that the allocated node will not be assigned to any other task and vice versa, even if the task is suspended or finished. Thus, a suspended task cannot be evicted to allow the hardware thread to be occupied by another task.
3. Tasks switch are not allowed. A running task holds the allocated hardware thread until the simulation ends.
4. The total number of created tasks (initial plus spawned tasks) is limited by the total number of processing nodes, that is, the number of cores multiplied by the number of threads. For example, a 2-core, 2-threaded system with one initial task is not allowed to spawn more than 3 additional tasks during execution, even after any of them finishes.

4.2 The Dynamic Scheduler

Processing element selection is done at run-time. Load balancing enhances the performance of the system. This approach is used with independent tasks.

The dynamic scheduler, implemented in Multi2Sim, offers a more flexible handling of software tasks, with the following criterion:

1. The mapping of initial tasks at startup does not differ from the static scheduler. However, these allocations are not definitive, and they can vary at runtime.
2. Tasks have a time quantum, specified as a number of cycles by the ContextQuantum variable. If an allocated task exceeds this quantum, and there is any unallocated task waiting for execution, it is selected for eviction by the dynamic scheduler.
3. New spawned tasks try to find a processing node that has not been used before by any other task, rather than choosing a processing node that was already allocated by any suspended or evicted task.

4. When an allocated task is suspended, it is immediately selected by the dynamic scheduler for eviction, so that any unallocated task waiting for execution can allocate the released processing node again.
   The raw Multi2Sim simulator used a fixed number of ContextQuantum. The result of changing this number is shown in Fig. 3. The result investigates that increasing ContextQuantum lead to increase system performance. When the ContextQuantum arrived at its half value, then The system performance starts to decrease.

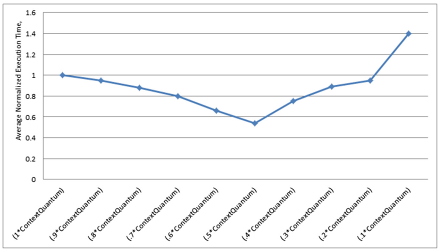


Fig. 3. ContextQuantum in dynamic scheduler.

In this experiment, we used varied benchmarks [19,20] as shown in Table 2. The tasks in these benchmarks are mixed between sequential tasks and parallel tasks. Each configure file has 8 cores and one thread pair core.

4.3 The Adaptive Scheduler

Previous experience has helped get the new idea of scheduling in Multi2Sim be called The Adaptive Scheduler, where the ContextQuantum is changed dynamically depending on the available free cores or free threads in Multi2Sim's processor. In other words, a quantum of ContextQuantum determines by FreeNodes.

$$\text{NewContextQuantum} = \text{ContextQuantum}/\text{FreeNodes}$$

Where Maximum FreeNodes = AllNode/2. For example, if the number of all nodes on your processor equal 8 nodes and free nodes equal 6, then the optimal new ContextQuantum will be equal NewContextQuantum = ContextQuantum/4.

5 Results

In this section, we first present a quantitative characterization of our design (adaptive scheduling) then comparing between diffident types of Task scheduler (static scheduling, dynamic scheduling)used in Multi2Sim simulator.

Tasks chosen in our study contain a mix of sequential and parallel applications. See Table 2 [19,20]. We generate different workload (task sets) consisting

Table 1. Multi-core configurations used in our study.

Configuration	Description
8 cores (one thread)	One thread per core so there are 8 nodes
8 cores (two threads)	Two thread per core so there are 16 nodes
8 cores (four threads)	Three thread per core so there are 32 nodes

Table 2. Benchmark applications.

Type	Suite	Inputs	Benchmarks
Sequential	SPEC2006	nput.program	bzip
		capture.tst	gobmk
		hyperviscoplastic.inp	calculix
		retro.hmm	hmm
		test.txt	sjeng
		lbm.in	lbm
		an4.ctl	sphinx
		inp.in	mcf
	MiBench	runme large.sh	basicmath
			bicount
			qsort
			susan
			dijkstra
			patricia
			sha
			adpcm
			fft
			gsm
			stringsearch
Parallel	PARSEC	simsmall	blackscholes
			swaptions
			canneal
			vips
			bodytrack

of varying mix of those applications. Across all the tasks sets, the ratio of sequential tasks ranges from 35% to 85%, so the ratio of parallel tasks ranges from 15% to 65%.

Figure 4 shows the execution time for Sequential and Parallel Tasks in Table 2. This figure shows that Adaptive scheduling gives the best result comparing to Static and Dynamic scheduling when applying Sequential and parallel tasks. But the improvement in the simulation result appears more clearly in case of Sequential tasks. That is because Parallel task does not need to migrate to another node. It mostly executed on the nodes which are assigned firstly.

5.1 Average Normalized Execution Time in Multi-core Task Scheduler

An inviting method of presenting processor performance is to normalize execution times to a reference processor, as it was done to obtain a SPEC ratio, and then take the average of the normalized execution times. However, if we average the normalized execution time values, the result will depend on the choice of the processor we used as a reference. In this paper, Static scheduling was taken as a reference. The Adaptive Scheduler gives Minimum normalize execution times Scheduling compared to the other schedulers (Static and Dynamic). Adaptive scheduling normalized execution time was 41% less than the corresponding Static scheduling normalized execution time. This normalized execution time refers to the average time to run tasks in the three used configuration see Table 1. The corresponding execution time in dynamic scheduling was 14% less than Static scheduling, See Fig. 5.

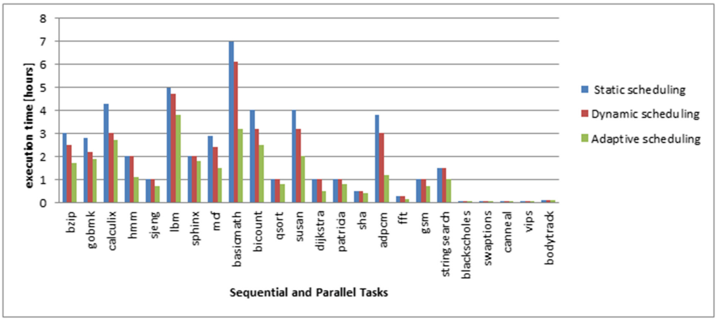


Fig. 4. Execution time in multi-core task scheduler.

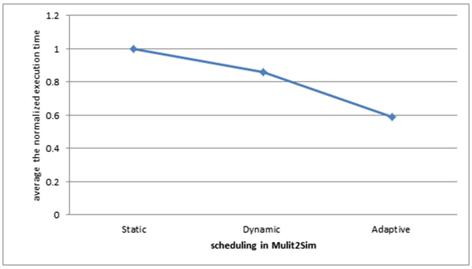


Fig. 5. Average normalized execution time in multi-core task scheduler.

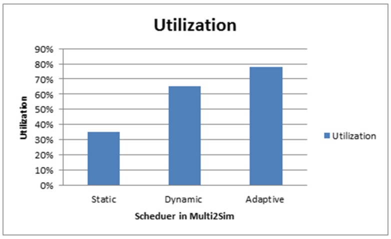


Fig. 6. Utilization in multi-core task scheduler.

5.2 Utilization in Multi-core Task Scheduler

Utilization in this context is the proportion of the total available processor cycles that are consumed by each process. Figure 6, reports the utilization (averaged across all tasks sets) for different configurations, where the best utilization (78%) in adaptive Scheduler. Dynamic Scheduler gives utilization equal to 65% while a very bad utilization resulted from using Static scheduling (35%).

6 Conclusions

In this paper, we presented Multi2Sim, a simulation framework that integrates important features of existing simulators and extends them to provide additional functionality. We try to improve this simulation by adding functionality and new task scheduler (Adaptive). The result shows improving in simulator performance (execution time and utilization) comparing to the traditional static and dynamic scheduler. As this tool has mainly research aims, it has been built to serve as support for future works, such as development and evaluation of performance improvement techniques. Multi2Sim is foreseen to be used both in the field of computer architecture and interconnection networks.

References

1. AMD Athlon^{TM} 64 X2 Dual-Core Processor Product Data Sheet, September 2006. www.amd.com
2. McNairy, M., Rohit Bhatia, R.B., Montecito, M.: A dual-core, dual-thread Itanium processor. IEEE Micro **25**(2), 10–20 (2005)
3. Kalla, R., Sinharoy, B., Tendler, J.M.: IBM Power5 chip: a dual-core multithreaded processor. IEEE Micro **25**(2), 40–47 (2005)
4. Shirazi, B.A., Kavi, K.M., Hurson, A.R.: Scheduling and Load Balancing in Parallel and Distributed Systems. IEEE Computer Society Press, Los Alamitos (1995)
5. Casavant, T.L., Kuhl, J.G.: A taxonomy of scheduling in general-purpose distributed computing systems. IEEE Trans. Softw. Eng. **14**(2), 141–154 (1988)
6. Ubal, R., Jang, B., Mistry, P., Schaa, D., Kaeli, D.: Multi2Sim: a simulation framework for CPU-GPU computing. In: Proceedings of the 21st International Conference on Parallel Architectures and Compilation Techniques, pp. 335–344. ACM (2012)

7. Ubal, R., Sahuquillo, J., Petit, S., Lopez, P.: Multi2sim: a simulation framework to evaluate multicore-multithread processors. In: IEEE 19th International Symposium on Computer Architecture and High Performance computing, pp. 62–68. Citeseer (2007)
8. Burger, D.C., Austin, T.M.: The simple scalar tool set, Version 2.0. Technical report CS-TR-1997-1342 (1997)
9. Tullsen, D.M.: Simulation and modeling of a simultaneous multithreading processor. In: 22nd Annual Computer Measurement Group Conference, December 1996
10. Moudgill, M., Bose, P., Moreno, J.: Validation of Turandot, a fast processor model for microarchitecture exploration. In: IEEE International Performance, Computing, and Communications Conference (1999)
11. Moudgill, M., Wellman, J., Moreno, J.: Environment for power PC microarchitecture exploration. IEEE Micro **19**(3), 15–25 (1999)
12. Magnusson, P.S., Christensson, M., Eskilson, J., Forsgren, D., Hallberg, G., Hogberg, J., Larsson, F., Moestedt, A., Werner, B.: Simics: a full system simulation platform. IEEE Comput. **35**(2), 50–58 (2002)
13. Marty, M.R., Beckmann, B., Yen, L., Alameldeen, A.R., Xu, M., Moore, K.: GEMS: multifacets general execution-driven multiprocessor simulator. In: International Symposium on Computer Architecture (2006)
14. Binkert, N.L., Hallnor, E.G., Reinhardt, S.K.: Network-oriented full-system simulation using M5. In: 6th Workshop on Computer Architecture Evaluation using Commercial Workloads (CAECW), Ref. 107, February 2003
15. Ubal, R., Sahuquillo, J., Petit, S., Lopez, P.: A simulation framework to evaluate multicore-multithreaded processors. In: Proceedings of the 19th International Symposium on Computer Architecture and High Performance Computing, Gramado, Brazil, October 2007
16. Zhao, X., Ma, S., Chen, W., Wang, Z.: Exploiting parallelism in the simulation of general purpose graphics processing unit program. J. Shanghai Jiaotong Univ. (Science) **21**(3), 280–288 (2016)
17. Gong, X., Ubal, R., Kaeli, D.: Multi2Sim Kepler: a detailed architectural GPU simulator. In: IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), pp. 269–278. IEEE (2017)
18. The Multi2Sim Simulation Framework. http://www.multi2sim.org
19. MiBench Benchmark. http://vhosts.eecs.umich.edu/mibench
20. SPEC CPU Benchmarks. http://www.spec.org/benchmarks.html

PFastNCA: Parallel Fast Network Component Analysis for Gene Regulatory Network

Dina Elsayad(✉), A. Ali, Howida A. Shedeed, and M. F. Tolba

Faculty of Computer and Information Sciences, Ain Shams University, Cairo, Egypt
{dina.elsayad,Dr_Howida}@cis.asu.edu.eg,
ahmedali@fcis.asu.edu.eg, fahmytolba@gmail.com

Abstract. One of the gene expression data analysis tasks is the Gene regulatory network analysis. Gene regulatory network is concerned in the topological organization of genes interactions. Moreover, the regulatory network is important for understanding the normal cell physiology and pathological phenotypes. However, the main challenge facing gene regulatory network algorithms is the data size. Where, the algorithm runtime is proportional to the data size. This paper presents a parallel algorithm for gene regulatory network (PFastNCA) which is an improved version of FastNCA. PFastNCA enhanced the main core of FastNCA which is the connectivity matrix estimation using a distributed computing model. Where, the work is divided among N processing nodes, PFastNCA is more efficient than FastNCA. It also achieved a better performance and speedup reached 1.91.

Keywords: Microarrays · Bioinformatics · Component analysis
High performance · Parallel · Regulatory network

1 Introduction

One of the recent important research areas is Bioinformatics. Bioinformatics can be defined as the application of information technology in the field of molecular biology. The Information technology application in the field of molecular biology includes management, processing and analysis of both molecular biological and genomic data. The main goal of Bioinformatics is to increase and develop the biological processes understanding [1]. Bioinformatics includes many research areas; according to Srikanth Aluru (2006), the major research areas in Bioinformatics are: Computational evolutionary biology [2], Gene expression analysis [3], Sequence analysis [4], Genome annotation [5], Protein expression analysis [6], Analysis of regulation [7], Protein-protein docking [8], Predictions of protein structure [9], analysis of mutations in cancer [10], Modeling biological systems [11], Comparative Genomics [12], High throughput image analysis [13] and Microarrays [14].

Microarray is one of the vital research fields in bioinformatics. It is a multiplex lab-on-a-chip that assays large amounts of biological material using high throughput screening methods [15]. The type of this biological material determines the type of microarrays. One of the microarrays is DNA microarrays [16].

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 585–594, 2018.
https://doi.org/10.1007/978-3-319-74690-6_57

The DNA microarray is a high throughput experimental procedure that measure expression levels of massive numbers of genes simultaneously. The amount of gene mRNA is an estimator for the gene expression level. Where, the more mRNA indicates more gene activity. Furthermore, the gene is transcribed if and only if it is active [17]. The DNA microarrays are a beneficial technique for gene expression analysis, disease gene identification and new diagnostic tool development. The microarrays data analysis is a big challenge because the data is huge and the analysis process involves many computational tasks [18]. The microarrays data analysis starts with the data sub-set extraction. During this step the differentially expressed genes (discriminator genes) are extracted. Moreover, the differentially expressed genes (output of this step) can be the input of the other analysis tasks such as clustering and network analysis [19]. The gene regulatory network [20] aims to study the genes interactions topological organization. The recent gene regulatory network analysis techniques can be classified into four main categories: Component analysis techniques [21–28], reverse engineering techniques [29], the regression techniques [30] and mutual information based techniques [31, 32]. In fact, almost all of these techniques are time consuming and computationally intensive. Therefore, parallel algorithms are needed to enhance the techniques performance.

The rest of the paper is organized as follows; the next section provides the needed background and the related work. Section 3 provides the proposed algorithm PFastNCA (Parallel Fast Network Component Analysis algorithm). The implementation and results are discussed in Sect. 4. Finally Sect. 5 provides the conclusion and future work.

2 Background and Related Work

The gene regulatory network [19] is one of the microarrays data analysis computational tasks that aims to study the gene interaction topological order and how the genes influence each other. As shown in Fig. 1 the gene regulatory network consists of a set of nodes and sets of edges, where the nodes represent genes and the edges represent regulatory interactions between genes. The illustrated example represents a regulatory network of Transcription Factor (TF) family of gene p53 in mouse. Where, Ellipses represent TFs and boxes represent genes. In addition, the hexagons are the clustered genes. Furthermore, the number of the genes is shown inside the hexagons. Moreover, the red lines indicate that protein-DNA binding is known.

As shown in Fig. 2 the gene regulatory network analysis techniques can be classified into four categories: Mutual information based techniques, regression techniques, reverse engineering techniques and component analysis techniques. Moreover, the component analysis technique can be further categorized into a number of sub-categories. The component analysis technique sub-categories are: Principal Component Analysis (PCA) [21], Singular value decomposition [22, 23], Independent component analysis [24, 26] and Network component analysis.

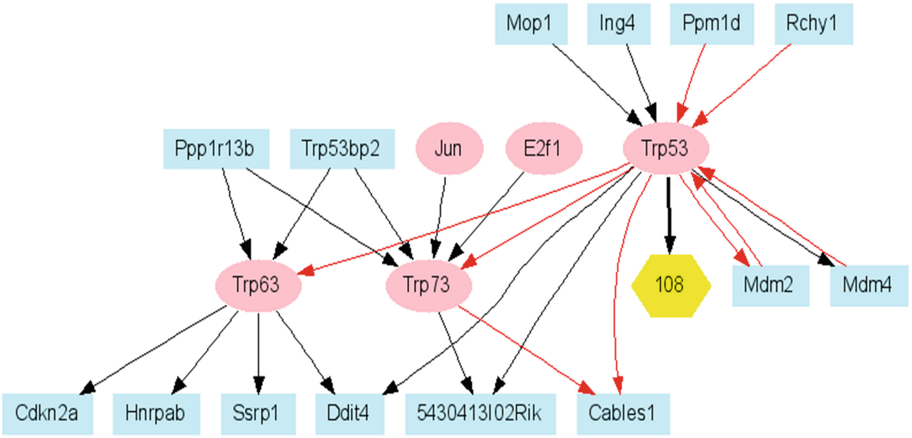


Fig. 1. Gene regulatory network of TF family p53 in mouse Image Source: http://rulai.cshl.edu/TRED/GRN/p53.htm.

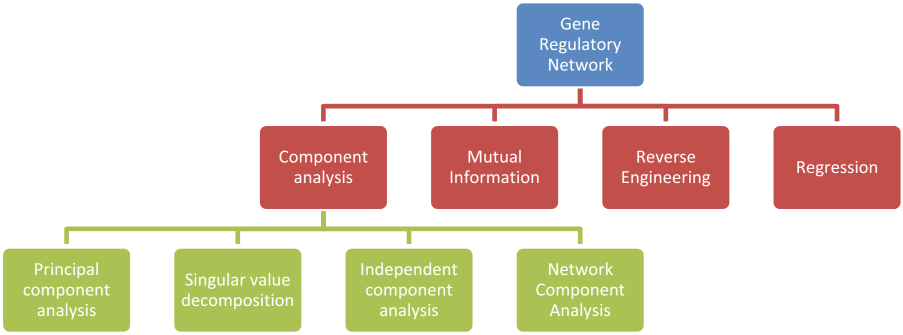

Gene Regulatory Network
Component analysis
Mutual Information
Reverse Engineering
Regression
Principal component analysis
Singular value decomposition
Independent component analysis
Network Component Analysis

Fig. 2. Gene regulatory network techniques

The principal component analysis technique (PCA) [21] is a dimensionality reduction technique that determines the key variables which clarify the difference in the data and simplify the data analysis as well as the data visualization. In other words, when applying PCA on a data matrix [A] of M observation in N variables; in the case of the gene expression data, the observation are the genes and the variables are the experiment conditions/samples, it finds new × variables which reduce the dimensionality of [A], where, the new × variables must be orthogonal and mutually uncorrelated. Also, it must take into consideration covering as much as possible the variance in the original variables. These variables are called the principle components. The principle component is a linear combination of the original variable.

The singular value decomposition (SVD) [22, 23] aims to discover the underlying patterns in the gene expression data. Where, the complex gene expression profiles can be captured by a small number of the characteristic modes that capture the temporal change pattern of gene expression data. Before applying the SVD, the rows and columns must have a zero mean. This is achieved by subtracting the mean values of the data and performing iterative normalization procedures for the rows and columns [33]. Given the gene expression data matrix [A] the SVD states that the singular values (denoted by s_i) are the square root of the Eigen-values of [A^T A].

Another gene regulatory network analysis technique is the independent component analysis (ICA) [24, 26]. The ICA is an unsupervised exploratory analysis technique that models the gene expression data matrix [A] using hidden variables [Y] as expressed in Eq. 1. The values of [X] matrix are the new “independent component” variables. The statistical dependences between the columns of matrix [X] must be minimized. Where, the statistical dependences between variables is computed by mutual information (MI). The ICA shows high fraction values around zero; therefore it can identify the approximate sparse components. To avoid this limitation Hyvarinen presented the FastNCA algorithm [28]. Where, the hidden variables [Y] (expressed in Eq. 2) are the product of the R and $C^{1/2}$. The $C^{1/2}$ is the linear correlation of the data covariance matrix C [34]. The rotation matrix R is initialized to random values and adjusted to minimize the statistical dependence between the independent components. The ICA suffers from data sensitivity particularly the regulatory variables whose influence of the genes follows the Super Gaussian distribution [26].

$$A = XY \tag{1}$$

$$Y = RC^{1/2} \tag{2}$$

The gene regulatory network is driven by hidden regulatory signals. Hence, the goal of the gene regulatory network analysis technique is to uncover these hidden signals. These hidden regulatory singles are: Transcription Factors (TFs) and are proteins involved in the processes of converting or transcribing DNA into RNA. The traditional computational method such as PCA and ICA ignore the underlying network structure and the decomposition is based on statistical constraints. Hence the provided model may not contain biologically or physically meaningful signals. Furthermore, these dimensional reduction techniques don’t address the hidden dynamic reconstruction problems. In addition, PAC and ICA restrict the data to be mutually orthogonal and statistically independent. To avoid these limitations Liao et al. presented the Network Component Analysis (NCA) technique [27]. The mathematical model is illustrated in Eq. 3 where [A] is the data matrix of size (M × N) and [X] is the connectivity matrix of size (M × L) which encodes the connectivity strength (transcription regulation strength) between regulatory signals (TFs) and the output domain (gene expression data). In addition, the single matrix [Y] is of size (L × N) where L is the number of the regulatory signals which is much smaller than M. The decomposition of [A] into two matrices [X] [Y] is an inverse problem which has no unique solution. Therefore, further assumptions are needed. These assumptions are called NCA criteria which are as follows:

- The connectivity matrix [X] must have a full column rank
- The connectivity matrix [X] must still have a full row rank when a node is removed which implies that [X] must have L-1 zeros
- The single matrix [Y] must have a full row rank

The NCA is an iterative algorithm where the initial guess of the connectivity matrix [X] is formed by setting all the elements corresponding to missing edges between the regulatory layer and the output layer to zero, moreover, the remaining elements are initialized to an arbitrary value. In addition, it suffers from computational instability and multiple local solution problems. Therefore, Change et al. presented the FastNCA (Fast Network Component Analysis) algorithm [28]. FastNCA is faster than NCA since it is a non-iterative algorithm; furthermore it avoids the NCA limitations. The model of the gene regulatory network is shown in Eq. 1, while, Eq. 3 illustrates the model with noise. The FastNCA algorithm is illustrated in Fig. 3.

$$B = A + N = XY + N \tag{3}$$

```
Input     [A] : Data matrix
Output    Gene Regulatory Network
Step 1:   Perform rank-L EYM (Eckart–Young–Mirsky) approximation of $B$ by SVD
Step 2:   Let W = $U_L$ (left singular vector)
Step 3:   Estimate connectivity matrix [$X$]
Step 3.1:     For i=1 to L
                  Re-order rows of W such that the i^th column of $X$ has the form $a_i = \begin{bmatrix}\tilde{a_i}\\ 0\end{bmatrix}$
                  Partition W = $\begin{bmatrix}W_c\\ W_r\end{bmatrix}$
                  Get [ $V_0$] the right singular vectors by applying SVD to $W_r$
                  Compute $\tilde{a_i}$ = the left singular vector of $W_c V_0 V_0^T$
Step 4:   Estimate the [$Y$] matrix
```

Fig. 3. FastNCA algorithm

3 PFastNCA

The gene expression data analysis is important for understanding the gene relationships and functions. The gene regulatory network is one of the gene expression data analysis tasks. The gene regulatory network goal is to study the genes interactions and how the genes influence each other. Furthermore, the gene regulatory network is vital for understanding the normal cell physiology and the pathological phenotypes. Unfortunately, most of the recent gene regulatory network techniques are affected and are of a massive data size. Therefore, this paper presents an algorithm for a gene regulatory network by using a high performance technique to enhance both the efficiency and speedup. The proposed algorithm is called PFastNCA (Shown in Figs. 4 and 5). PFastNCA stands for Parallel Fast Network Component Analysis. It is a parallel version of the FastNCA algorithm [28] presented by Chunq Chang, etc. This proposed algorithm is part of the PAGeneRN framework [35]. In other words, the PFastNCA technique is part of the data analysis module of the PAGeneRN framework.

```
Input       [A] : Data matrix
Output      Gene Regulatory Network
Step 1:     Perform rank-L EYM (Eckart–Young–Mirsky) approximation of $B$ by SVD
Step 2:     Let W = $U_L$ (left singular vector)
Step 3:     Estimate connectivity matrix [$X$] in parallel
Step 3.1:       Send chunk of the data for each processing node
Step 3.2:       Collect data from the processing nodes
Step 4:     Estimate the [$Y$] matrix
```

Fig. 4. FastNCA parallel version – master node

```
Input       [W] : Left singular vector matrix
Output      Connectivity sub matrix [$X$]
Step 1:     Parallel for i=1 to L
                Re-order rows of W such that the $i^{th}$ column of $X$ has the form $a_i = \begin{bmatrix}\tilde{a_i}\\0\end{bmatrix}$
                Partition W = $\begin{bmatrix}W_c\\W_r\end{bmatrix}$
                Get [ $V_0$] the right singular vectors by applying SVD to $W_r$
                Compute $\tilde{a_i}$ = the left singular vector of $W_c\ V_0\ V_0^T$
Step 2:     Send connectivity sub matrix [$X$] to the master node
```

Fig. 5. FastNCA parallel version – worker node

From the data distribution model perspective; the parallelism can be functionality or data parallelism. In the functionality parallelism; different independent functions are done by different processing nodes simultaneously. On the other hand, in the data parallelism; the data is distributed among the processing nodes where each processing node processes parts from the data. In other words, in the data parallelism technique usually each process takes an N/P data item where N is the number of data items and P is the number of processing nodes. The proposed algorithm uses the data parallelism technique, where the master node will distribute the data among the worker nodes, where each work node will take a data chunk of the size N/P.

From hardware architecture; the parallelism can be a shared memory model or distributed model. On one hand, in the shared memory model the processing nodes share the same memory. On the other hand, in the distributed memory model each processing node has its own memory and the nodes communicate with each other by sending/receiving messages. Furthermore, the two models can be combined together to benefit from the available resources. This model is called a hybrid model where each processing node has its memory (as in distributed model). Furthermore, each node has work items that share the same memory (as in shared memory model). The proposed algorithm uses the distributed model. Where the data is distributed among the P processing nodes; each node takes an N/P data item. Furthermore, the master node gathers the results back from the P processing nodes.

4 Implementation and Results

This section demonstrates some experiments to evaluate the performance of the proposed algorithm PFastNCA against the original FastNCA algorithm. The PFastNCA is implemented using C++ with MPI. While, FastNCA is implemented using C++. The experiments were conducted on a 36 processing nodes cluster where each node is Intel® Xeon® CPU E5620 @ 2.40 GHZ. For comparison six large microarrays datasets are used, one of them are ovarian cancer and the other datasets are breast cancer. These datasets are publicly available from the GEO database (http://www.ncbi.nlm.nih.gov/) through their accession numbers. Table 1 shows the accession number and the size (number of genes × number of samples) of each dataset.

Table 1. Microarrays datasets used for comparison

DataSet no.	Accession no.	Size
1	GSE6008	22283 × 104
2	GSE7390	22283 × 189
3	GSE2034	22283 × 256
4	GSE3494	22645 × 252
5	GSE9195	54675 × 78
6	GSE6532	54675 × 88

Table 2 indicates the overall exestuation time of PFastNCA and FastNCA. Where, PFastNCA was running using 36 processing nodes and overall estimation time is measured for all the datasets (in seconds). Furthermore, the results of measuring the overall estimation time of PFastNCA using different numbers of processing nodes is depicted in Fig. 6. The Figure shows that PFastNCA is more efficient than FastNCA providing better performance and reaching an average speed-up of 3.81. Figure 6 shows the overall execution time of PFastNCA for the sixth dataset (accession no. GSE6532) using different number of the processing nodes. The results indicate that PFastNCA is more efficient than FastNCA providing better performance and reaching the average speedup of 1.34.

Table 2. Algorithm overall execution time (in seconds)

Dataset no.	1	2	3	4	5	6
FastNCA	72.590	80.430	95.166	102.192	215.102	219.117
PFastNCA	38.178	42.235	57.189	66.711	159.368	168.130
Speedup	1.901	1.910	1.664	1.531	1.350	1.304

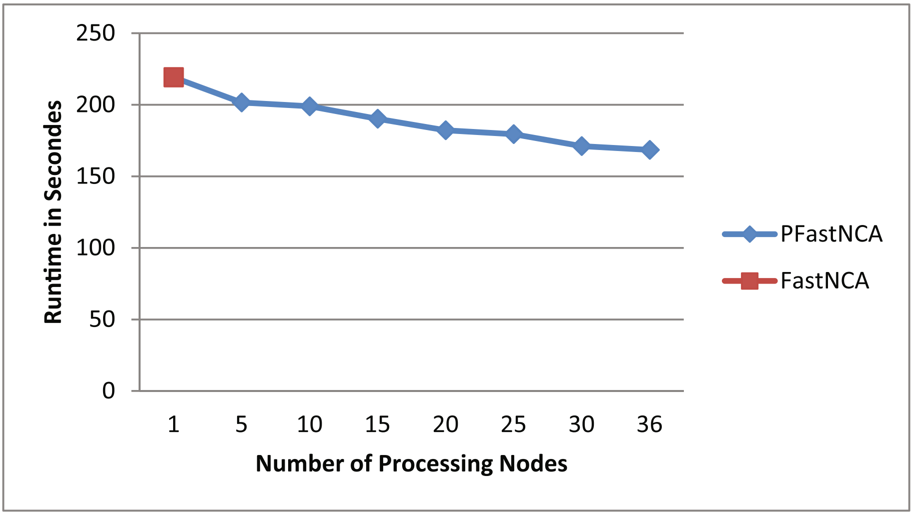


Fig. 6. Runtime for data set GSE6532

5 Conclusion and Future Work

One of the microarrays data analysis tasks is the gene regulatory network which aims to study the gene-to-gene interactions and how the genes influence each other. The gene regulatory network is vital for understanding the normal cell physiology and the pathological phenotypes. One of the gene regulatory network techniques is the FastNCA algorithm which is a dimensionality reduction algorithm that identifies the key variables in the dataset. PFastNCA is an improved version of FastNCA that enhanced the most computational consuming task of FastNCA which is the connectivity matrix estimation using a distributed computing model. In the distributed computing model, the data is divided among N processing nodes. PFastNCA successfully reduced the runtime as well as the speedup of the FastNCA algorithm. Using a number of microarrays data sets, the experimental results showed that PFastNCA outperformed FastNCA algorithm runtime. The measured speedup factor reached 1.91. For future work we aim to enhance the performance of another algorithm of gene regulatory network algorithms and compare between different gene regulatory network techniques categories.

References

1. Nair, A.: Computational biology & bioinformatics - a gentle overview. Commun. Comput. Soc. India **30**(1), 7–12 (2007)
2. Cosmides, L., Tooby, J.: From Function to Structure: The Role of Evolutionary Biology and Computational Theories in Cognitive Neuroscience. The MIT Press, Cambridge (1995)
3. Velculescu, V.E., Zhang, L., Vogelstein, B., Kinzler, K.W.: Serial analysis of gene expression. Science **270**(5235), 484–487 (1995)

4. Durbin, R.: Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids. Cambridge University Press, Cambridge (1998)
5. Kelley, L.A., MacCallum, R.M., Sternberg, M.J.: Enhanced genome annotation using structural profiles in the program 3D-PSSM. J. Mol. Biol. **299**(2), 501–522 (2000)
6. Ghaemmaghami, S., Huh, W.-K., Bower, K., Howson, R.W., Belle, A., Dephoure, N., O'Shea, E.K., Weissman, J.S.: Global analysis of protein expression in yeast. Nature **425**(6959), 737–741 (2003)
7. Janssen, P.J., Jones, W.A., Jones, D.T., Woods, D.R.: Molecular analysis and regulation of the glnA gene of the gram-positive anaerobe Clostridium acetobutylicum. J. Bacteriol. **170**(1), 400–408 (1988)
8. Dominguez, C., Boelens, R., Bonvin, A.M.: HADDOCK: a protein-protein docking approach based on biochemical or biophysical information. J. Am. Chem. Soc. **125**(7), 1731–1737 (2003)
9. Shortle, D.: Prediction of protein structure. Curr. Biol. **10**(2), 49–51 (2000)
10. Berrozpe, G., Schaeffer, J., Peinado, M.A., Real, F.X., Perucho, M.: Comparative analysis of mutations in the p53 and K-ras genes in pancreatic cancer. Int. J. Cancer **58**(2), 185–191 (1994)
11. Haefner, J.W.: Modeling Biological Systems: Principles and Applications. Springer, US (2005)
12. Rubin, G.M., Yandell, M.D., Wortman, J.R., Gabor, G.L., Nelson, C.R., Hariharan, I.K., Fortini, M.E., Li, P.W., Apweiler, R., Fleischmann, W.: Comparative genomics of the eukaryotes. Science **287**(5461), 2204–2215 (2000)
13. Dowsey, A.W.: High-Throughput Image Analysis for Proteomics. Citeseer (2005)
14. Churchill, G.A.: Fundamentals of experimental design for cDNA microarrays. Nat. Genet. **32**(1), 490–495 (2002)
15. Culf, A., Cuperlovic-Culf, M., Ouellette, R.: Carbohydrate microarrays: survey of fabrication techniques. OMICS J. Integr. Biol. **10**(3), 289–310 (2006)
16. Gasch, A., Spellman, P., Kao, C., Carmel-Harel, O., Eisen, M., Storz, G., Botstein, D., Brown, P.: Genomic expression programs in the response of yeast cells to environmental changes. Mol. Biol. Cell **11**(12), 4241–4257 (2000)
17. Schena, M., Shalon, D., Davis, R., Brown, P.: Quantitative monitoring of gene expression patterns with a complementary DNA microarray. Science **270**, 467–470 (1995). Washington
18. Yang, Y., Choi, J., Choi, K., Pierce, M., Gannon, D., Kim, S.: BioVLAB-Microarray: microarray data analysis in virtual environment. In: IEEE Fourth International Conference on eScience (2008)
19. Haman, J., Valenta, Z.: Shrinkage approach for gene expression data analysis. EJBI **9**(3), 2–8 (2013)
20. Aluru, S.: Handbook of Computational Molecular Biology. CRC Press, Boca Raton (2006)
21. Raychaudhuri, S., Stuart, J.M., Altman, R.B.: Principal components analysis to summarize microarray experiments: application to sporulation time series. In: Pacific Symposium on Biocomputing, pp. 455–466. NIH Public Access (2000)
22. Watkins, D.S.: Fundamentals of Matrix Computations, vol. 64, pp. 309–409. John Wiley & Sons, Chichester (2004)
23. Holter, N.S., Mitra, M., Maritan, A., Cieplak, M., Banavar, J.R., Fedoroff, N.V.: Fundamental patterns underlying gene expression profiles: simplicity from complexity. Proc. Natl. Acad. Sci. **97**(15), 8409–8414 (2000)
24. Hyvarinen, A., Karhunen, J., Oja, E.: Independent Component Analysis. John Wiley & Sons, New York (2001)

25. Aapo, H.: Fast and robust fixed-point algorithms for independent component analysis. IEEE Trans. Neural Netw. **10**(3), 626–634 (1999)
26. Liebermeister, W.: Linear modes of gene expression determined by independent component analysis. Bioinformatics **18**(1), 51–60 (2002)
27. Liao, J.C., Boscolo, R., Yang, Y.-L., Tran, L.M., Sabatti, C., Roychowdhury, V.P.: Network component analysis: reconstruction of regulatory signals in biological systems. Proc. Natl. Acad. Sci. **100**, 15522–15527 (2003)
28. Chang, C., Ding, Z., Hung, Y.S., Fung, P.C.W.: Fast network component analysis (FastNCA) for gene regulatory network reconstruction from microarray data. Bioinformatics **24**(11), 1349–1358 (2008)
29. Jostins, L., Jaeger, J.: Reverse engineering a gene network using an asynchronous parallel evolution strategy. BMC Syst. Biol. **4**(1), 17–33 (2010)
30. Gregoretti, F., Belcastro, V., Di Bernardo, D., Oliva, G.: A parallel implementation of the network identification by multiple regression (NIR) algorithm to reverse-engineer regulatory gene networks. PLoS ONE **5**(4), e10179–e10183 (2010)
31. Sales, G., Romualdi, C.: parmigene—a parallel R package for mutual information estimation and gene network reconstruction. Bioinformatics **27**(13), 1876–1877 (2011)
32. Shi, H., Schmidt, B., Liu, W., Muller-Wittig, W.: Parallel mutual information estimation for inferring gene regulatory networks on GPUs. BMC Res. Notes **4**(1), 189–194 (2011)
33. Eisen, M.B., Spellman, P.T., Brown, P.O., Botstein, D.: Cluster analysis and display of genome-wide expression patterns. Proc. Natl. Acad. Sci. **95**(25), 14863–14868 (1998)
34. Walpole, R.E., Myers, R.H., Myers, S.L., Ye, K.: Probability and Statistics for Engineers and Scientists. Macmillan, New York (1993)
35. Elsayad, D., Ali, A., Shedeed, H.A., Tolba, M.F.: PAGeneRN: parallel architecture for gene regulatory network. In: Handbook of Research on Machine Learning Innovations and Trends, pp. 762–786. IGI Global (2017)

Robots That Can Mix Serious with Fun

Ibrahim A. Hameed(✉), Girts Strazdins, Håvard A. M. Hatlemark,
Ivar S. Jakobsen, and John O. Damdam

Department of ICT and Natural Sciences, Faculty of Information Technology and Electrical Engineering, Norwegian University of Science and Technology (NTNU),
Larsgårdsvegen 2, 6009 Ålesund, Norway
ibib@ntnu.no

Abstract. In this paper, we propose the use of a robot platform for helping teachers and students in classrooms and at homes without degrading the academic achievement. We hypothesized that the robot can teach the same contents without degrading the academic achievement. In addition, it can improve the quality of teaching, mixing teaching with fun to attract students' attention, repeat the contents without fatigue or anger and also providing pupils with special needs with the right contents in the right time. In this paper a 24 degrees of freedom (DOF) NAO robot is used for teaching an introduction to robotics to elementary school pupils. A GUI tool is designed to help teachers to easily upload slides of the subject's contents. The robot presents the slides by reading it while generating random movements that resemble the body language of human presenters and teachers. The robot uses its camera to count the number of faces heading to it and use it as a measure of attention. When the number goes below a threshold number, the robot refreshes the classroom by automatically switching from teaching mode into entertainment mode where it dances, sings, tells a story, tells jokes, etc. in a manner that to attracts the attention of the pupils. In an experiment, pupils from stages 5 and 6 are divided into two groups; one group attended the same class content with the IT teacher and the second group attended with the robot. A questionnaire is designed to evaluate pupils' knowledge before and after attending the class. Results showed that the robot could attract students and provide them with more deeply understanding of technical terms. As a future work, different subjects will be used and more experiments will be conducted.

Keywords: Teaching · Education · Learning · Social robots
Educational robots

1 Introduction

The use of robotics to support various industries has increased remarkably during the past decade. The use of robotics in an educational context has become a popular topic [1–4]. In a study, the impressions of the audience for an educational robot presenter was compared to those of a computer-animated presenter, revealed that the robot resulted in enhanced impressions on audiences [5]. Thus, the presence of physical robots is expected to be useful as aids in many interactive fields including entertainment [6], education [7, 8], security [9],

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 595–604, 2018.
https://doi.org/10.1007/978-3-319-74690-6_58

rescue [10] and elderly care [11]. Various studies had showed that technologies could be used to enhance student motivation [12]. Therefore, educational robots can be considered a step in the evolution of educational technology, and numerous models have been successfully implemented in educational applications [13]. Previous studies have revealed that educational robots can communicate effectively and enhance student enjoy [13–15]. Increasingly more studies have incorporated educational robots as tools for supporting educational activities, because they offer new benefits in educational environments. In a study, it was observed that there was positive response from children with robot-aided tuition where the robot enhanced their motivation [16].

In this paper, a teaching assistance robot is designed to help a classroom teacher to present slides in an interactive and enjoyable way to attract student attention and motivate them. A user-friendly graphical user interface (GUI) is designed to allow the classroom teacher to upload slides so the robot can read and run it while mimicking the motion of human presenters. The robot is enabled to predict students' focus and switch modes from teaching to entertainment in a manner that keep students attracted. The robot uses natural languages processing (NLTK) to build a user profile for each student in order to use call him/her by name to achieve what is called for long-term interaction with its users [17–19]. The paper is organized as follows: an introduction is presented in Sect. 1. In Sect. 2, materials and methods are presented in details. Experiment setup and results are presented in Sect. 3. In Sect. 4, concluding remarks are drawn.

2 Material and Method

2.1 NAO Robot

Nao, shown in Fig. 1, is Aldebarans first humanoid robot. Nao was first introduced in 2006. Nao is standing tall at 58 cm and is under continuously development. NAO is currently on his 5th version. At this point there has been sold over 7,000 NAOs throughout the world. NAO is an endearing, interactive and personalizable robot companion. Everyone can construct his own experience with specific applications based on his own imagination and needs [20]. The fruit of a unique combination of mechanical engineering and software, NAO is a character made up of a multitude of sensors, motors and software piloted by a made-to-measure operating system: NAOqi OS. NAO has seven senses for natural interaction described as follows:

Moving: 25 degrees of freedom and a humanoid shape that enable him to move and adapt to the world around him. His inertial unit enables him to maintain his balance and to know whether he is standing up or lying down.

Feeling: The numerous sensors in his head, hands and feet, as well as his sonars, enable him to perceive his environment and get his bearings.

Hearing and speaking: With his 4 directional microphones and loudspeakers, NAO interacts with humans in a completely natural manner, by listening and speaking.

Seeing: NAO is equipped with two cameras that film his environment in high resolution, helping him to recognize shapes and objects.

Connecting: To access the Internet autonomously, NAO is able to use a range of different connection modes (WiFi, Ethernet).

Thinking: We can't really talk about "Artificial Intelligence" with NAO, but the robots are already able to reproduce human behavior (Aldebaran-Robotics 2016).

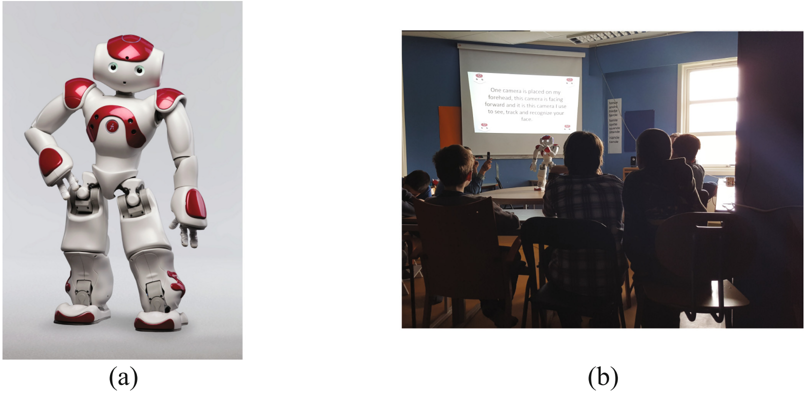

(a) (b)

Fig. 1. NAO robot (a) and NAO presentation to students of stages 5 and 6 at Ålesund International School (AaIS).

2.2 Long-Term Interaction with Robots

According to [21], to building up child-robot relationship from initial attraction towards long-term social engagement, the robot should be able to present more human skills (e.g. verbal and non verbal communication skills, motor competences and assertive tasks). A natural way to build on this conclusion is to use vocal interaction to try and create a bond between the robot and a human.

2.3 Natural Language Toolkit (NLTK)

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum [17]. NLTK is used for information extraction from natural language in digital sentences and text as databases.

2.4 Speech Recognition

Speech recognition is defined as the ability of devices to respond to spoken commands. Speech recognition enables hands-free control of various devices and equipment (a particular boon to many disabled persons), provides input to automatic translation, and creates print-ready dictation. Python has a Speech recognition API that is able to convert speech to text using a dedicated Library for performing speech recognition, with support for several engines and APIs, online and offline [22]. In addition, there is also a cloud speech API developed by Google [23].

2.5 Face Detection

Face detection is the recognition of human faces in digital pictures done by computing. Several companies develop face recognition software and market it as a product, including Cognitec, Ayonix, Looksery, Google and Omron.

2.6 Social Interaction

Speech recognition, shown in Fig. 2, can be carried out in various ways: (1) using NAOqi: where NAO robot has its own on-board speech recognition library (i.e., ALSpeechRecognition) that compares a pre-determined vocabulary to the sound input working internally on Nao, and (2) using online Google Speech Recognition engine where a sound file is recorder and sent to Google's online speech recognition API and retrieve the text [23]. Google's speech recognition is found to be of better accuracy but at the cost of network time delay. When the text returns from Google's Speech Recognition engine, only important key information is extracted though the use of a keyword extractor. Keyword extractor extracts important information that can be used for building/updating a user profile.

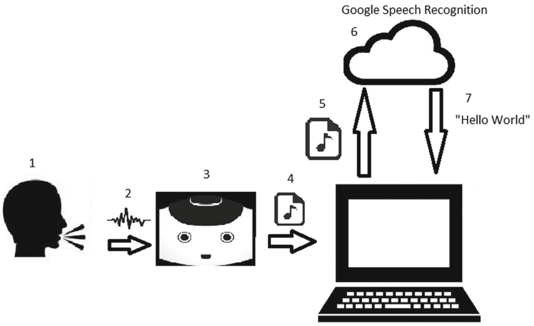


Fig. 2. Speech recognition in NAO: (1) a person speaks, (2) sound waves is transmitted, (3) Nao robot records sound waves, (4) recorded sound waves stored inside Nao and Sent to a local computer, (5) the computer sends the sound file to Google Speech Recognition API, (6) Google Speech Recognition checks sound file and converts it to text, and finally (7) a text string is generated and then sent back to local computer.

2.7 User Profiles

The profiles for the users are stored locally on the computer as objects stored by a cPickle on the project directory. The profiles are stored as a dictionary list in cPickle, where the key to the dictionary is the users name and the value is the whole profile object. The user profiles contain the user's picture and basic personal information and other keywords obtained from the interaction. The robot uses facial detection and recognition where a picture of the user is obtained and compared to pictures stored in the user profiles directory. If the robot can detect the person, it returns its name and other information to be used in the dialogue system so the user profile can be further updated. If not, the robot constructs a new user profile for that new user.

2.8 NAO Control Center (NCC)

A front-end application to easily upload PowerPoint slides is developed, as it is shown in Fig. 3. The resulting GUI, also called Nao Control Center (NCC), implements some features inspired by the Choregraphe software, such as the volume slider, Auto, Sleep and Wake Up buttons. The Auto button sets the NAOqis "autonomous life" - behavior state to "solitary", making the Nao robot stand up and turn its head toward faces or sounds. The Sleep button sets the NAOqis "autonomous life" - behavior states to "disabled", making the Nao robot sits down and turn off the motor stiffness. The Wake Up button makes the Nao robot go to a standing pose and turn on motor stiffness, if disabled. In the bottom there is a status line displaying information regarding the last action on the NCC. When browsing for files, either for presentations or user profile, only suitable files will be visible (.pptx and .txt).

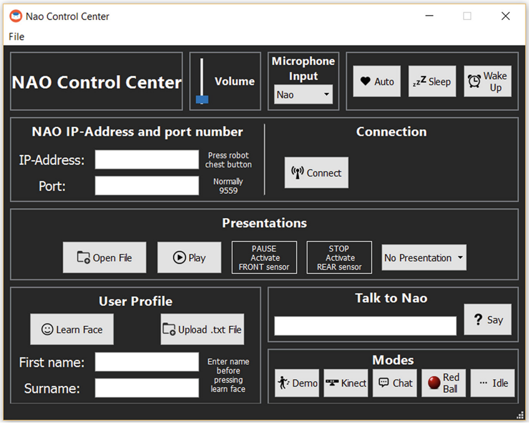


Fig. 3. Nao Control Center (NCC) graphical user interface (GUI).

2.9 Lecturer Robot

The developed Nao robot as a lecturer or teaching assistant uses its features, mainly talking, to teach about a subject. A PowerPoint presentation runs in the background working as subtitles for the Nao robot, including some images and videos for some specific lectures. In this way, the audience will get both audio and visual input about the subject. The function to pause the lecture is linked to the Nao robots front head sensor, if activated the presentation will come to a halt when the active slide is completed. To continue, reactivate the front head sensor. To stop a presentation, activate the rear head sensor on the Nao robot, and it will stop and close PowerPoint when the current slide is completed.

2.10 Robot Animation and Movements

The custom dances and movements is made with Choreographs timeline block and extracted to python code with Choreographs “extract with Bezier” function. Using Choreographs ‘Animation mode’ it is possible to move the robot to the desired posture and save it in a time slot. This saves a lot of time compared to calculating every single angle of joints in order to achieve such complex movements. The Bezier option makes the movements more smooth and natural. In NAOqi there is over 600 pre made animations called behaviors in ‘ALBehaviorManager’. Some of them are just a second long and some are as long as half a minute. The ‘Behaviors’ class in the application contains methods to easily start a portion of these behaviors. The ‘movements’ class contains the custom made movements, dances and the motor control method for the Kinect mode. The Nao robot will by standard implemented functions adjust to some degree the movements, to avoid colliding with its own body and limbs. During lecturing, the robot will be allowed to mimic the random movements of human presenters.

3 Results

In this section experiment setup and results are presented.

3.1 Experiment Setup

The experiment is conducted with guidelines from NSD - Norwegian Centre for Research Data, for research including people under the age of 18. It is registered in their project database as project number 48068. The survey is conducted on a sample of eleven pupils in fifth and sixth grade in elementary school. The collaborating teacher is an IT teacher for the sample pupils in the same school. The survey population will take a pre-lecture questionnaire about the functions of a robot, and be asked to draw a picture of a humanoid robot. After the pre-lecture questionnaire, pupils will be divided into two groups. One group, six pupils, three at a time, will see the Nao robot giving a lecture. The other group, five pupils, first three of them followed by the last two, will get a lecture from the IT teacher on the same topic. Both lectures are about the Nao robot’s specifications, abilities and function including how and why it works. The Nao robot uses the “Nao presentation” from the front-end application. The teachers PowerPoint presentation is keywords, data and the same images as the “Nao presentation”, but the teacher has to formulate it in its own way to convey the information. After the lecture, the surveyed pupils will be asked to answer a post lecture questionnaire, with the same questions and a drawing task as the pre-lecture. The survey population will formulate their own written answer to each question. Answers will be assessed based on certain answer keys for each question, ignoring misspelling and synonyms. Pre- and post-lecture questions and favorable answer keys are giving below:

1. How do robots see us? What are their eyes? Keyword: Camera
2. How do robots hear us? What are their ears? Keyword: Microphone
3. How do robots speak? What is their mouth? Keyword: Speaker
4. What do robots use to move? What are their muscles? Keyword: Motor
5. How can robots avoid bumping into things? How do they sense obstacles? Keyword: Sensor
6. How can robots think? What is their brain? Keyword: Computer
7. How can humans instruct a robot to do a specific task? Keyword: Programming or coding
8. How can a robot be connected to other robots? Keyword: Any name or description of a cabled or wireless network

The post-lecture questionnaire contains an added question different for the groups. Robot lecture recipients are asked “Did you like having the robot in the classroom?” The teacher group is asked, “What do you think a lesson would be if it were given by a robot?”

3.2 Experiment Results

Going through the same material, although, the robot have more slides compared to the teacher (65 vs. 12), Nao robot used only about half the time the teacher did on the lecture, as it is shown in Table 1. Figure 4 shows the amount of correct answer keys for the 8 questions of both the pre-and the post-lecture. It is obvious that students who attended for the robot gained more deep understanding of the topic. Figure 5 shows how the answers to “Did you like having the robot in the classroom?” given to the robot group post-lecture. Possible answers were multiple choices with the options: “Not at all”, “Nothing special”, “Don’t know”, “Yes”, and “Yes, it was super-exiting”. Figure 6 shows what kind of robot the pupils drew. The number of pre-lecture drawing of robots does not match the number of participants because some drew more than one robot, and all drawn robots are included. In the post-lecture, everyone drew a drawing resembling the Nao robot, even the teacher group who had only seen images of it.

Table 1. Number of slides and time used by NAO and IT teacher.

	NAO	Teacher
Number of slides	65	12
Time used (min)	12	20–25

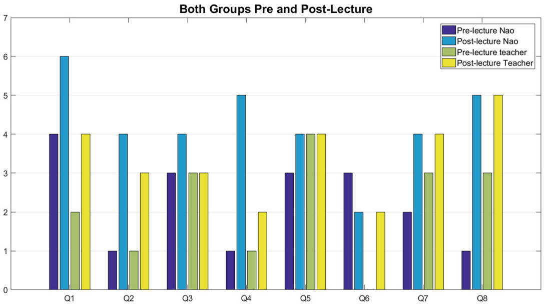


Fig. 4. The number of correct keys contained in the answer of each of the eight questions for the Nao and teach groups before and after the experiment.

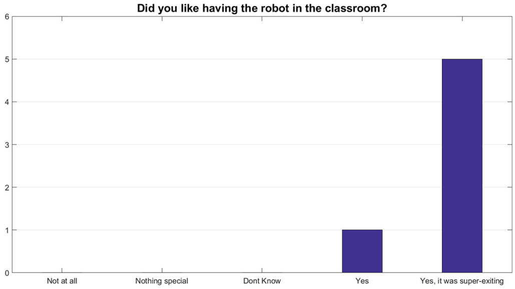


Fig. 5. Answers of post-lecture Nao group to the question "Did you like having the robot in the classroom?".

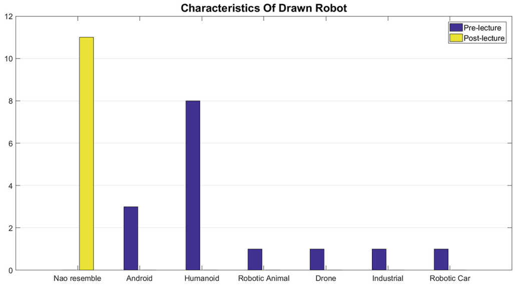


Fig. 6. Generic type of robot drawn before and after the lecture.

4 Conclusions

It is not expected that a robot could replace a human teacher soon. A teacher is a human with a non perfect memory, and in elementary school they often teach in several subjects with aging textbooks as sources for knowledge. The base of this experiment was to use a robot as an assistant to the classroom's teacher where it can efficiently look for information and present most relevant ones to students. The robot also can be used to present subjects and repeat it in a customized way without fatigue. The ability to mix serious teaching with fun can be used to improve educational atmosphere and make it attractive for students. In the experiment presented in this paper, the sample size was too small to make any representative statistics, and is rather used to make a proof of concept. One of the most significant findings is the time consumption, the teacher used approximately twice the time the robot used for teaching the same content. One reason is that the robot was giving a one-way lecture with no input or feedback from students. The robot managed to hold the attention for the complete lecture with boost to excitement and some laughter from the jokes. The teacher presentation held a more serious tone throughout the lecture, while the pupils interacted with raising hands and asking questions when they felt the need.

Two observers were observing the level of focus, noise, and excitement of the pupils during the robot and the teacher presentations. Regarding focus, the students in all classes were very much paying attention to both the robot the teacher presentations. Level of noise were to some degree higher in the class with the teacher; because of interaction, asking questions, pupils came with suggestions on how things works. The pupils were quite more excited in the robot classroom due to robot dancing, telling jokes, playing music and doing some fantastic acrobatic arts by letting the robot stands on one foot. The teacher created excitement through; interaction, asking the pupils questions, facilitate an atmosphere where the pupils were very engaged and they came with suggestion on how things are working. To come closer to a significant result from the tests, and make sure it is not a random result. There should be more groups, and each group should receive more lectures. It is reasonable so suspect that pupils can have a different attitude towards the robot over time (with more lectures and due to technology's effect). Therefore it is necessary to carry out a higher number of presentations with the robot with the same groups.

References

1. Chang, C.-W., Lee, J.-H., Wang, C.-Y., Chen, G.-D.: Improving the authentic learning experience by integrating robots into the mixed-reality environment. J. Comput. Educ. **55**(4), 1572–1578 (2010)
2. Ryu, H.J., Kwak, S.S., Kim, M.S.: Design factors for robots as elementary school teaching assistants. J. Bull. Jpn. Soc. Sci. Des. **54**(6), 39–48 (2008)
3. Goldstain, O.H., Ben-Gal, I.E., Bukchin, Y.: Evaluation of telerobotic interface components for teaching robot operation. IEEE Trans. Learn. Technol. **4**(4), 365–376 (2011)
4. Belghith, K., Nkambou, R., Kabanza, F., Hartman, L.: An intelligent simulator for telerobotics training. IEEE Trans. Learn. Technol. **5**(1), 11–19 (2012)
5. Nishimura, Y., Kushida, K., Dohi, H., Ishizuka, M., Takeuchi, J., Nakano, M., Tsujino, H.: Development of multimodal presentation markup language MPML-HR for humanoid robots and its psychological evaluation. J. Humanoid Robot. **4**(1), 1–20 (2007)
6. Bretan, M., Cicconet, M., Nikolaidis, R., Weinberg, G.: Developing and composing for a robotic musician using different modes of interaction. In: Proceedings of the International Computer Music Conference (ICMC), pp. 498–503, Ljubljana, Slovenia (2012)
7. Han, J.-H., Jo, M., Park, S., Kim, S.: The educational use of home robots for children. In: Proceedings of IEEE Robot Human Interactive Communication, pp. 378–383 (2005)

8. Benitti, F.B.V.: Exploring the educational potential of robotics in schools: a systematic review. J. Comput. Educ. **58**(3), 978–988 (2012)
9. Song, G., Yin, K., Zhou, Y., Cheng, X.: A surveillance robot with hopping capabilities for home security. IEEE Trans. Consum. Electron. **55**(4), 2034–2039 (2009)
10. Murphy, R.R.: Human-robot interaction in rescue robotics. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **34**(2), 138–153 (2004)
11. Wada, K., Shibata, T., Saito, T., Tanie, K.: Effects of robot-assisted activity for elderly people and nurses at a day service center. Proc. IEEE **92**(11), 1780–1788 (2004)
12. Serio, A.D., Ibáñez, M.B., Kloos, C.D.: Impact of an augmented reality system on students' motivation for a visual art course. J. Comput. Educ. **68**, 586–596 (2013)
13. Cooper, M., Keating, D., Harwin, W., Dautenhahn, K.: Robots in the classroom-tools for accessible education. In: Proceedings of the 5th European Conference Advancement of Assistive Technology, pp. 448–452 (1999)
14. You, Z.-J., Shen, C.-Y., Chang, C.-W., Liu, B.-J.: A robot as a teaching assistant in an English class. In: Proceedings of the 6th International Conference of Advanced Learning Technologies, pp. 87–91 (2006)
15. Chang, C.-W., Lee, J.-H., Chao, P.-Y., Wang, C.-Y., Chen, G.-D.: Exploring the possibility of using humanoid robots as instructional tools for teaching a second language in primary school. J. Educ. Technol. Soc. **13**(2), 13–24 (2010)
16. Lee, E.K., Lee, Y.J.: A pilot study of intelligent robot aided education. In: Proceedings of the 16th International Conference of Computer Education, pp. 595–596 (2008)
17. Natural-Language-Toolkit. Nltk 3.0 documentation. http://www.nltk.org/. Accessed 21 May 2016
18. Hameed, I.A.: Using natural language processing (NLP) for designing socially intelligent robots. In: 2016 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob), Cergy-Pontoise, pp. 268–269 (2016)
19. Hameed I.A., Tan, Z.-H., Thomsen, N.B., Duan, X.: User acceptance of social robots. In: Proceedings of the Ninth International Conference on Advances in Computer-Human Interactions (ACHI 2016), Venice, Italy, pp. 274–279 (2016)
20. Aldebaran-Robotics: Who is NAO? (2016). https://www.aldebaran.com/en/cool-robots/nao. Accessed 21 May 2016
21. Han, J., Kim, D.: r-Learning services for elementary school students with a teaching assistant robot. In: 2009 4th ACM/IEEE International Conference on Human-Robot Interaction (HRI), La Jolla, CA, pp. 255–256 (2009)
22. Python Speech recognition 3.4.3. https://pypi.python.org/pypi/SpeechRecognition/3.4. Accessed 21 May 2016
23. Cloud speech API: Speech to text conversion powered by machine learning. https://cloud.google.com/speech/. Accessed 25 Sept 2017

Dialect Versus MSA Sentiment Analysis

Sandra Rizkallah[1](✉), Amir Atiya[1], Hossam ElDin Mahgoub[2],
and Momen Heragy[2]

[1] Faculty of Engineering, Computer Engineering Department,
Cairo University, Giza, Egypt
sandrawahid@hotmail.com
[2] AlKhawarizmy Software, Cairo, Egypt
http://www.alkhawarizmy.com

Abstract. The growth of social media has made Arabic sentiment analysis an active research area. The challenges lie in the fact that most users write unstructured dialect texts instead of writing in Modern Standard Arabic (MSA). In this paper we address these challenges by comparing between two strategies: applying sentiment analysis algorithms directly on the dialect; and applying a translation that transforms from dialect to MSA, then designing a sentiment analysis on the resulting MSA text. We consider Saudi Twitter data.

Keywords: Sentiment analysis · Modern standard arabic
Twitter · Arabic language · Saudi dialect

1 Introduction

Arabic Sentiment Analysis has gained much attention these days due to the increasing number of Arabic texts on different social media platforms such as Twitter. The main challenge that we face is that social media communicate mostly with dialect, not with Modern Standard Arabic (MSA). This fact limits standardizing sentiment analysis techniques, to deal directly with any Arabic text due to the plethora of dialects and their changing nature. Moreover, applying standard machine learning techniques directly on dialect may have questionable performance in comparison to applying them on MSA. This is due to the variations of Arabic dialects and the more randomness in their structure of the sentence.

In this work we address this problem, and investigate whether it is better to apply sentiment analysis directly to the Arabic dialect, or to apply a dialect-MSA translator, and then use MSA based sentiment analysis tools. The disadvantage of the former is the aforementioned difficulty in dealing with the variations in dialect. The disadvantage of the latter is the addition of one extra translation step, which could introduce some performance loss.

On the other hand, the latter will make use of the extensive resources and work done on MSA. In summary, the contributions of this work include the following:

- Compare sentiment analysis methods on the dialect directly, versus on MSA after applying a translation.

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 605–613, 2018.
https://doi.org/10.1007/978-3-319-74690-6_59

- Investigate the performance of the well-known AlKhawarizmy dialect-to-MSA translator [9, 10].
- Apply machine learning methods for sentiment analysis on Saudi-dialect tweets (a corpus of 2010 tweets).
- Compare between seven machine learning models for the sentiment analysis task.

We review here previous work on sentiment analysis for Arabic dialects.

In [1] the authors test the effect of preprocessing on the performance of sentiment analysis. This is done by collecting an Egyptian dialect dataset from Twitter. In [2] the authors introduce the largest sentiment analysis dataset to-date for the Arabic language. Baseline experiments are run on the dataset to establish a benchmark for both sentiment polarity classification and rating classification. In [3] an Arabic Twitter data set is collected and manually annotated. The data set includes a mixture of Arabic dialects. In [4] an Arabic social sentiment analysis dataset is gathered from Twitter. The data set focuses on Egyptian dialect. The experiments done provide benchmark results for 4-way sentiment classification. In [5] sentiments toward a specific topic are extracted from Saudi dialect tweets. Sentiment analysis is applied combining lexicon and supervised approaches. In [6] the authors present a Saudi dialect Twitter corpus for sentiment analysis. In [7] the authors address the challenge of tweeting in dialectical Arabic. They propose a hybrid approach where a lexical-based classifier will label the training data, which is fed to the SVM machine learning classifier. In [8] an Arabic language dataset of opinions on health services is collected from Twitter. Sentiment analysis is applied on the collected dataset.

2 Data Preparation

2.1 Data Collection and Annotation

The data used for the experiments done is a collection of 2010 Saudi dialect tweets. We have manually annotated these tweets to take one of the following four labels to represent the sentiment of the tweet:

1. Positive
2. Negative
3. Neutral
4. Mixed.

2.2 Data Preprocessing

Preprocessing on the tweets data has to be done to be able to manipulate this data. The following are the preprocessing operations we have carried out:

- Removing URL's and user mentions
- Removing English words and English punctuation
- Removing Arabic punctuation
- Removing Arabic stop words; in this step negation detection is performed not to remove a negation word as a stop word.

- Removing “Tatweel” for example: “العـــــربية” is transformed to “العربية”
- Removing redundant repetition of characters for example: “سععععععععودي” is transformed to “سعودي”
- Handling emoticons by transforming the emoticon symbol to an indicative Arabic word for example: “:D” is transformed to “ضحكة”
- Handling hashtags by removing the underscore character and splitting the hashtag into the words it constitutes for example: “#السعودية_اليوم” is transformed to 2 words “السعودية” and “اليوم”.

3 Dialect to MSA Translation

The main goal of this module is to translate from Saudi dialect tweets to MSA tweets. This goal is achieved by tokenizing each tweet then getting the MSA translation of each token using Social Analytics dynamic-link library (dll) from “AlKhawarizmy Software”.

Figure 1 outlines the sentiment analysis comparison done with and without using the dialect to MSA translation module.

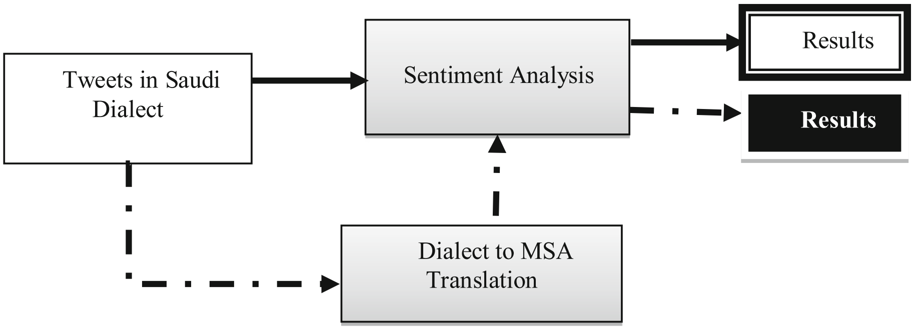


Fig. 1. A brief outline of the work done

We have designed a translation algorithm such that we benefit from the capabilities of the dll as much as possible. This can be elaborated as follows:

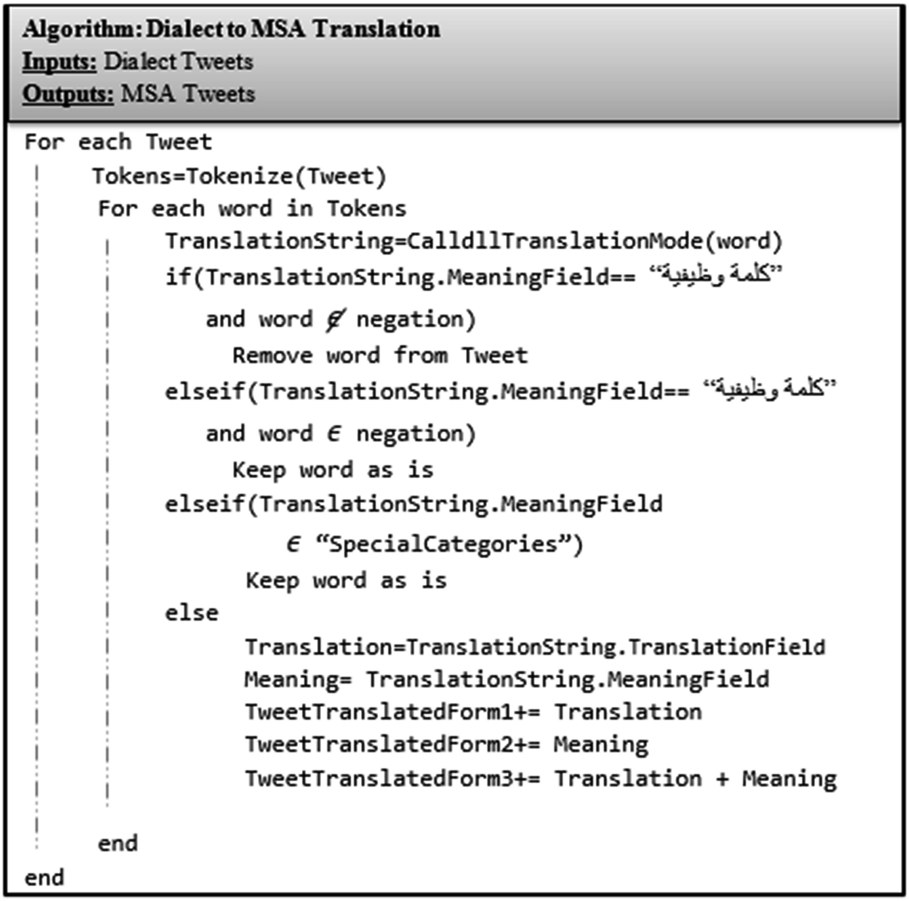
- Used the meaning fields associated with the word’s translations in the dll to check for special words that may belong to any of 79 “Special Categories” such as:

  “اسم شخص” , “الدول”, “العواصم”, “الجماعات”, “الأدوات”, “ مصطلح حاسوبي”,
  “الأوقات”, “المذاهب السياسية”, “الأنهار”, “الألقاب”, “وحدات الزمن”

  and others.
- Used the meaning field “كلمة وظيفية” to remove more stop words than those removed from the preprocessing step. Making one more check that the word is not a negation word.

- Establishing 3 forms of translations for each tweet:
  1. Using Translation field of dll
  2. Using Meaning field of dll
  3. Using Translation field of dll followed by Meaning field of dll.

The algorithm designed for the dialect to MSA translation module is as follows:



Algorithm: Dialect to MSA Translation
Inputs: Dialect Tweets
Outputs: MSA Tweets

```
For each Tweet
    Tokens=Tokenize(Tweet)
    For each word in Tokens
        TranslationString=CalldllTranslationMode(word)
        if(TranslationString.MeaningField== "كلمة وظيفية"
           and word ∉ negation)
              Remove word from Tweet
        elseif(TranslationString.MeaningField== "كلمة وظيفية"
           and word ∈ negation)
              Keep word as is
        elseif(TranslationString.MeaningField
                 ∈ "SpecialCategories")
              Keep word as is
        else
              Translation=TranslationString.TranslationField
              Meaning= TranslationString.MeaningField
              TweetTranslatedForm1+= Translation
              TweetTranslatedForm2+= Meaning
              TweetTranslatedForm3+= Translation + Meaning

    end
end
```

After translating Tweets to MSA, we now have tweets that follow MSA language rules, so we can apply stemming procedures to restore the word to its stem. We have used the ISRI Arabic Stemmer [11] that shares many features with the Khoja stemmer [12].

4 Supervised Sentiment Analysis

4.1 Feature Extraction

The features extracted are n-gram of texts where an n-gram is a contiguous sequence of n items from a given sequence of text. We have extracted: unigrams, bigrams and trigrams.

Moreover, term frequency-inverse document frequency (tf-idf) transform is used to reflect the importance of a word to a document in a corpus. The tf-idf value increases proportionally to a word's frequency in a document, but is affected by the word's frequency in the corpus to adjust for the fact that some words appear more frequently in general.

4.2 Classifiers and Tuning

For the classifiers used, we have done k-fold cross-validation on the training data in order to tune the classifier's parameters. The classifiers used include: Logistic Regression classifier where the parameter tuned is the inverse of regularization strength, Passive Aggressive classifier [13] where the aggressiveness parameter is tuned, Support Vector Machine (SVM) where the penalty parameter of the error term is tuned, Perceptron where the parameter tuned is the number of passes over the training data (aka epochs), Multinomial Naive Bayes classifier (MNB) where the additive smoothing parameter is set to the universal default 1, Stochastic Gradient Descent (SGD) [14] where the parameters tuned are the initial learning rate and the number of passes over the training data (aka epochs) and finally K-Nearest Neighbors (KNN) where the parameter tuned is the number of neighbors (k).

4.3 Experiments

We have done various classification experiments on the data referred to in Sect. 2 (2010 tweets). The main goal of the experiments is to compare the classification results obtained from the dialect tweets to those obtained from the translated tweets. We have used 60% of the data as training data and 40% of the data as test data. All the experiments are applied using the different features referred to in Sect. 4.1. Moreover, the experiments are applied using the different classifiers referred to in Sect. 4.2.

Experiment 1: D
In this experiment, the data used is the tweets in the Saudi dialect form. These tweets are fed to the preprocessing module before performing classification.

Experiment 2: T
In this experiment, the data used is the tweets in the Saudi dialect form. These tweets are fed to the preprocessing module then the generated tweets are fed to the translation module to obtain Translated tweets of form 1 (using the translation field).

Experiment 3: M
In this experiment, the data used is the tweets in the Saudi dialect form. These tweets are fed to the preprocessing module then the generated tweets are fed to the translation module to obtain Translated tweets of form 2 (using the meaning field).

Experiment 4: TM
In this experiment, the data used is the tweets in the Saudi dialect form. These tweets are fed to the preprocessing module then the generated tweets are fed to the translation module to obtain Translated tweets of form 3 (using the translation field followed by the meaning field).

Experiment 5
In this experiment, a majority vote algorithm is developed to obtain the results based on the majority result of:

- Experiment 2 with weight 2
- Experiment 3 with weight 1
- Experiment 4 with weight 1.

Experiment 6
In this experiment, a majority vote algorithm is developed to obtain the results based on the majority result of:

- Experiment 1 with weight 2
- Experiment 2 with weight 2
- Experiment 3 with weight 1
- Experiment 4 with weight 1.

5 Results and Discussion

The following tables show the results of the experiments done using various features with various classifiers. The performance evaluation is done using two key metrics:

- F1-score:
  Also known as F-score or F-measure. The F1 score is the harmonic average of the precision *p (the number of correct positive results divided by the number of all positive results)* and recall *r (the number of correct positive results divided by the number of positive results that should have been returned.)*, where an F1 score reaches its best value at 1 and worst at 0. It is computed as follows:

$$2 * \frac{Precision * Recall}{Precision + Recall} \tag{1}$$

- Accuracy:
  It is the number of correct predictions made divided by the total number of predictions.

$$\frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \tag{2}$$

The numbers in Tables 1 and 2 represent F1-score and accuracy % for different experiments and classifiers.

Table 1. Results using Features 1-gram, 2-gram and 3-gram

Feature: 1-gram, 2-gram and 3-gram												
Versions	Dialect D		Translation T		Meaning M		Translation followed by Meaning TM		Majority Vote (T weight = 2, M weight = 1, TM weight = 1)		Majority Vote (D weight = 2, T weight = 2, M weight = 1, TM weight = 1)	
Logistic Regression	**0.67**	**72.4%**	**0.71**	**75.0%**	**0.70**	**73.9%**	**0.71**	**74.4%**	**0.70**	**74.1%**	**0.71**	**74.6%**
Passive Aggressive	0.69	73.1%	0.72	75.8%	0.71	73.9%	0.71	74.1%	0.71	74.3%	0.72	75.1%
SVM	**0.68**	**72.8%**	**0.71**	**74.3%**	**0.72**	**75.0%**	**0.72**	**75.0%**	**0.70**	**73.8%**	**0.70**	**74.1%**
Perceptron	0.68	69.7%	0.72	73.4%	0.70	70.4%	0.69	70.2%	0.71	73.0%	0.71	73.4%
Multinomial NB	0.70	74.4%	0.70	74.4%	0.70	73.1%	0.70	73.4%	0.70	73.8%	0.70	74.4%
SGD	**0.70**	**73.5%**	**0.73**	**75.9%**	**0.70**	**73.0%**	**0.70**	**74.0%**	**0.73**	**75.3%**	**0.72**	**74.8%**
KNN	0.53	65.9%	0.57	67.3%	0.58	66.2%	0.58	66.0%	0.57	66.8%	0.55	66.5%

Table 2. Results using Features 1-gram, 2-gram and 3-gram with Tfidf

Feature: 1-gram, 2-gram and 3-gram with Tfidf												
Versions	Dialect D		Translation T		Meaning M		Translation followed by Meaning TM		Majority Vote (T weight = 2, M weight = 1, TM weight = 1)		Majority Vote (D weight = 2, T weight = 2, M weight = 1, TM weight = 1)	
Logistic Regression	**0.70**	**73.6%**	**0.74**	**76.2%**	**0.72**	**75.4%**	**0.73**	**75.5%**	**0.72**	**74.8%**	**0.72**	**75.0%**
Passive Aggressive	0.70	73.0%	0.74	76.6%	0.71	74.0%	0.71	73.5%	0.73	75.3%	0.72	75.0%
SVM	**0.70**	**73.6%**	**0.74**	**75.9%**	**0.72**	**74.1%**	**0.72**	**73.9%**	**0.72**	**74.6%**	**0.72**	**75.0%**
Perceptron	0.69	69.7%	0.71	71.5%	0.70	69.9%	0.70	69.8%	0.72	72.5%	0.72	73.4%
Multinomial NB	0.54	66.3%	0.52	65.6%	0.52	65.4%	0.52	65.4%	0.52	65.4%	0.53	65.7%
SGD	**0.71**	**74.0%**	**0.74**	**76.1%**	**0.72**	**74.4%**	**0.71**	**73.0%**	**0.73**	**75.1%**	**0.73**	**75.5%**
KNN	0.66	71.6%	0.66	70.7%	0.64	69.3%	0.64	68.7%	0.65	69.8%	0.66	71.8%

One can observe that using the AlKhawarizmy translator (from dialect to MSA) [9, 10], we achieve better results (than applying directly on the Saudi dialect). In particular, the second majority vote classifier produces consistently better performance in 6 or 7 out of 7 classifiers, and for all n gram feature combination, and all tfidf/no-tfidf options. This shows the benefit and value of the AlKhawarizmy translator [9, 10].

6 Conclusions

We have managed to prove practically that applying sentiment analysis techniques yields better results on MSA data than on dialect data. The experiments done are using Saudi dialect tweets. We have applied 4-way supervised classification using 7 classifiers: Logistic Regression, Passive Aggressive, SVM, Perceptron, Multinomial Naive Bayes, SGD and KNN. We have also designed a preprocessing module that cleans raw Arabic tweets data and handles: emoticons, Tatweel and redundant repetition of characters. Furthermore, we have designed a translation algorithm that translates from Saudi dialect to MSA using Social Analytics dynamic-link library (dll) from "AlKhawarizmy Software".

Acknowledgements. The authors would like to thank ITIDA [15] for the support of this research. We thankfully acknowledge Dr. Tarek Nabhan for supplying us with Twitter data.

References

1. Shoukry, A., Rafea, A.: Preprocessing Egyptian dialect tweets for sentiment mining. In: The Fourth Workshop on Computational Approaches to Arabic Script-based Languages, pp. 47–56 (2012)
2. Aly, M., Atiya, A.: LABR: a large scale Arabic book reviews dataset. In: Proceedings ACL, vol. 2, pp. 494–498 (2013)
3. Refaee, E., Rieser, V.: An Arabic twitter corpus for subjectivity and sentiment analysis. In: LREC, pp. 2268–2273 (2014)
4. Nabil, M., Aly, M., Atiya, A.: ASTD: Arabic sentiment tweets dataset. In: Proceedings EMNLP, pp. 2515–2519 (2015)
5. Alhumoud, S., Albuhairi, T., Alohaideb, W.: Hybrid sentiment analyser for Arabic tweets using R. In: 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K), vol. 1, pp. 417–424 (2015)
6. Assiri, A., Emam, A., Al-Dossari, H.: Saudi twitter corpus for sentiment analysis. Int. J. Comput. Electr. Autom. Control Inf. Eng. **10**(2), 272–275 (2016)
7. Aldayel, H.K., Azmi, A.M.: Arabic tweets sentiment analysis – a hybrid scheme. J. Inf. Sci. **42**(6), 782–797 (2016)
8. Alayba, A.M., Palade, V., England, M., Iqbal, R.: Arabic language sentiment analysis on health services. arXiv preprint arXiv:1702.03197 (2017)
9. AlKhawarizmy Software. http://alkhawarizmy.com/. Accessed 9 Oct 2017
10. ElDin Mahgoub, H., Shaaban, Y.: A translator for arabic dialects to modern standard Arabic. In: The International Workshop on Computers and Information Sciences (WCIS), At Tabuk, Kingdom of Saudi Arabia (2015)
11. NLTK documentation. http://www.nltk.org/_modules/nltk/stem/isri.html. Accessed 9 Oct 2017
12. Khoja, S.: Stemming Arabic Text (1999). http://zeus.cs.pacificu.edu/shereen/research.htm. Accessed 9 Oct 2017

13. Crammer, K., Dekel, O., Keshat, J., Shalev-Shwartz, S., Singer, Y.: Online passive-aggressive algorithms. J. Mach. Learn. Res. JMLR **7**, 551–585 (2006)
14. Zadrozny, B., Elkan, C.: Transforming classifier scores into accurate multiclass probability estimates. In: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 694–699 (2002)
15. ITIDA. http://www.itida.gov.eg/. Accessed 9 Oct 2017

Energy Aware Optimized Hierarchical Routing Technique for Wireless Sensor Networks

Nermeen M. Hamza[1(✉)], Shaimaa Ahmed El-said[1],
Ehab Rushdy Mohamed Attia[2], and Mahmoud Ibrahim Abdalla[1]

[1] Faculty of Engineering, Electronics and Communications Department,
Zagazig University, Zagazig, Egypt
eng.nermeen88@gamil.com, saelsaid@pnu.edu.sa

[2] Faculty of Computer and Informatics, Zagazig University, Zagazig, Egypt
ehab.rushdy@gmail.com

Abstract. Wireless Sensor Networks (WSNs) ordinarily be composed of a large number of low-power sensor nodes which having several functions, that are a battery powered, and thus have very limited energy capacity. To lengthen the operational lifetime of a sensor network, energy efficiency should be considered in every aspect of sensor network design. In this paper, Enhanced Hierarchical Routing Technique (EHRT) is proposed to overcome the constraint of limited energy capacity of sensor nodes which enhancing the network lifetime and the energy efficiency. The suggested technique is a cluster-based routing which optimizes the low-energy adaptive clustering hierarchy routing technique (LEACH) by using a modified artificial fish swarm algorithm (AFSA). This modified AFSA selects the optimum clusters' head (CHs) locations by applying a number of behaviors following, preying and swarming on each cluster separately and using a modified fitness function to compare these behaviors' outputs to select the best CHs locations for each cluster separately. A framework for evaluating the performance is constructed and applied to verify the efficiency of the suggested technique comparing to other energy efficient routing techniques; optimized hierarchical routing technique (OHRT), low-energy adaptive clustering hierarchy (LEACH), and particle swarm optimized (PSO) routing techniques. The proposed technique yields best results than other techniques OHRT, LEACH, and PSO in terms of energy consumption and network lifetime. It reduces the energy dissipation by factor 0.7 compared with OHRT.

Keywords: Wireless sensor networks (WSNs)
Cluster-based routing technique
Low energy adaptive clustering hierarchy routing technique (LEACH)
Artificial fish swarm algorithm (AFSA)
Enhanced hierarchical routing technique (EHRT)

1 Introduction

Wireless sensor networking is a rising technology which promises a wide range of potential applications in both civilian and military regions [1]. It typically comprises of a large number of low-cost, low-power, and multi-functional sensor nodes that are

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 614–623, 2018.
https://doi.org/10.1007/978-3-319-74690-6_60

deployed in an area of interest. Those sensor nodes are small in size, embedded microprocessors, and radio transceivers [2]. In many WSN applications, the deployment of sensor nodes is performed in an ad hoc fashion without careful pre-planning and engineering. Once deployed, the sensor nodes have to be able to unrestrictedly organize themselves into a wireless communication network [3].

There are a lot of clustering algorithms that are proposed for homogenous WSNs like LEACH [4] and Threshold Sensitive Energy Efficient sensor network protocol (TEEN) [5]. New bio-inspired algorithms [6] were developed to see if they can cope with the challenging optimization problems [3] like an Improved Artificial Fish Swarm Algorithm (IAFSA) [7], optimized Bio-Inspired Hybrid routing Protocol (BIHP) [8], Ant Colony Optimization algorithm (ACO) [9] and Optimized Hierarchical Routing Technique (OHRT) [10].

1.1 Problem Statement

Reducing power consumption is the most important objective in the design of a sensor network. Since sensor nodes are powered by a battery. In most cases, they are deployed in a hostile or harsh environment, where it is extremely difficult or even impossible to change or recharge the batteries, it is important to reduce the power consumption of sensor nodes so that the lifetime of the sensor nodes and the whole network is prolonged.

Routing and data dissemination is an essential issue in (WSNs). Protocols for those networks have to be designed in such a way that the limited power in the sensor nodes is used efficiently. Low-Energy Adaptive Clustering Hierarchy (LEACH) [4] is a dynamic clustering protocol which attracted a very high attention because for its simplicity, energy efficient, and load balancing properties. However, there exist a few disadvantages in LEACH as sensor nodes, with lower initial energy, that function as CHs for the equal number of rounds as other sensor nodes, with higher initial energy, will perish prematurely. This could make energy holes and coverage problems; Since CH election is performed as regards probabilities, it is difficult for the predetermined CHs to be uniformly distributed throughout the network.

1.2 Our Contribution

The aim of this paper is to improve hierarchical routing which aims to efficiently decrease the energy consumption of sensor nodes through including them in multi-hop communication within a specified cluster. The proposed technique uses modified AFSA [11] to select the optimal CHs considering residual energies of nodes and their distance from the base station (BS) and substituting the role of CHs among cluster members that balance energy consumption and save more energy in nodes.

2 Enhanced Hierarchical Routing Technique (EHRT)

The proposed EHRT depends on the principle of clustering algorithm using LEACH routing protocol which optimized by using a modified AFSA that modifies the fitness function of AFSA. The fitness function is modified such that the best CHs location for each cluster is selected separately from the cluster head outputs; the following, praying and swarming behaviors, instead of selecting a certain behavior for all clusters in networks. The suggested technique consists of two phases: clusters formation and CHs election phase, and optimal CHs selection phase.

2.1 Clusters Formation and CHs Election

In this phase, clusters are formed and the CHs are elected based on the principle of the clustering algorithm. Forming clusters are completed using equal segmentation of the area space. CH for each formed cluster is elected depend on the node that requires the minimum transmission energy As all sensor nodes have initially the same energy, one at the center of the cluster is selected to be CH as its center of joint nodes in our design. The cluster formation and CHs election processes (Fig. 1).

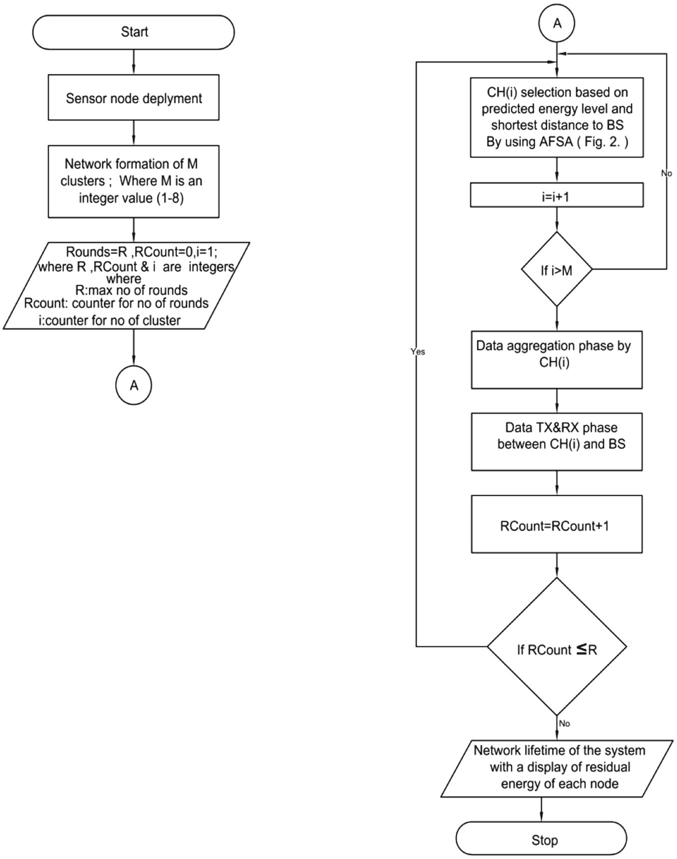

Start
Sensor node deplyment
Network formation of M clusters ; Where M is an integer value (1-8)
Rounds=R ,RCount=0,i=1; where R ,RCount & i are integers where R:max no of rounds Rcount: counter for no of rounds i:counter for no of cluster
A
A
CH(i) selection based on predicted energy level and shortest distance to BS By using AFSA (Fig. 2.)
No
i=i+1
If i>M
Yes
Data aggregation phase by CH(i)
Data TX&RX phase between CH(i) and BS
RCount=RCount+1
If RCount ≤R
No
Network lifetime of the system with a display of residual energy of each node
Stop

Fig. 1. Flowchart of the proposed optimized hierarchical routing technique

2.2 Optimal CHs Selection by Using AFSA

As CHs consumes high energy during data in gathering and transfer phase, EHRT rotates the CH role within the sensor nodes for each cluster by AFSA [12] at the beginning of every transmission round and thereby the energy wastage is being lessened and the energy utilization of each node is being maximized to protract the network lifetime. Figure 2 illustrates the CHs selection steps.

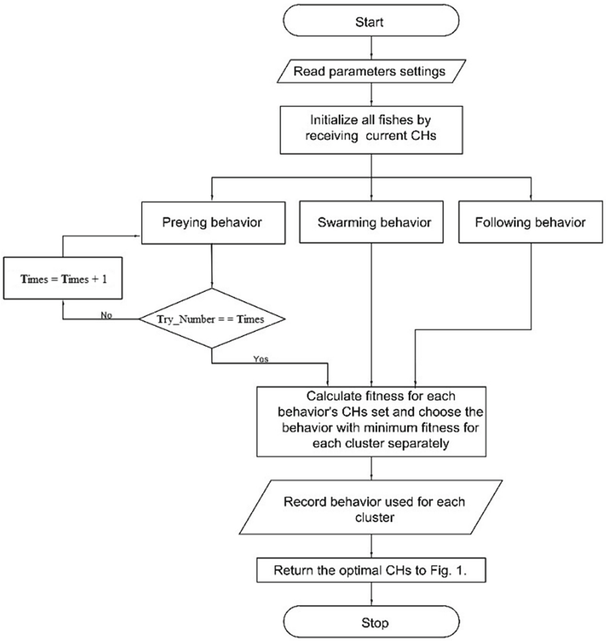

Start
Read parameters settings
Initialize all fishes by receiving current CHs
Preying behavior
Swarming behavior
Following behavior
Times = Times + 1
No
Try_Number = = Times
Yes
Calculate fitness for each behavior's CHs set and choose the behavior with minimum fitness for each cluster separately
Record behavior used for each cluster
Return the optimal CHs to Fig. 1.
Stop

Fig. 2. Flowchart of CHs selection by using AFSA

2.3 Evaluating the AFSA Behaviors

Each AFSA behavior produces a set of CHs. A proposed fitness function is used to select the best CHs. Equations (1), (2), (3) and (4) proposed to describe the fitness function, considering that fitness is calculated for each cluster. The smallest fitness represents the best CHs set among others such that the selected CH has the highest residual energy and the optimal location. The fitness function chooses the CHs set that gives the highest energy and the lowest distance to the base station (BS).

$$\text{Fitness} = \min\left[G_p(m), G_f(m), G_s(m)\right] \tag{1}$$

$$G_J(m) = \propto f_1(m) + (1 - \propto) f_2(m) \tag{2}$$

$$f_1(\mathrm{m}) = \frac{\sum_{\mathrm{i}=1}^{\mathrm{Nm}} \mathrm{E}\left(\mathrm{n_i}\right)}{\mathrm{E}\left(\mathrm{CH_m}\right)} \tag{3}$$

$$f_2(m) = \frac{d\left(n_{i,} CH_m\right)}{\left|C_m\right|} \tag{4}$$

Where m is the cluster number, $G_J(m)$ represents the fitness function using (J) behavior for a certain cluster m as $G_p(m)$ is preying behavior fitness, $G_f(m)$ is following behavior fitness and $G_s(m)$ is swarming behavior fitness. Referring to Eq. 1, the selected fitness for a certain cluster m is the lowest fitness behavior $G_J(m)$ which calculated in Eq. 2. In Eq. 3, $f_1(m)$ represents energy part for a certain cluster m and is equal to the sum of member node energy $E\left(n_i\right)$ in such cluster (not including CH) divided by the CH energy $E\left(CH_m\right)$ in that cluster m. Good CHs set gives the smallest f_1 value. While $f_2(m)$ represents the density of cluster m, it is equal to the average distance between CH and joined member nodes $d\left(n_{i,} CH_m\right)$ divided by the total member nodes in the same cluster C_m as in Eq. 4. Good CHs set should give the smallest f_2 value. This ensures that the CHs are close to their neighbors and the BS. The smallest fitness means the best CHs.

3 Simulation Results

To prove the proposed technique efficiency, its performance is evaluated and compared to other energy efficient routing techniques. For our experiments; Fig. 3 shows simulation environment of 250 sensor nodes that were deployed randomly at distance region of 300 m × 300 m between (x = 0; y = 0) and (x = 300; y = 300) with a fixed base station at location (x = 0, y = 0). The covered network is then clustered into 2, 3, 4, 5, 6, 7, 8 quadrants Fig. 4 illustrates the covered network clustered of seventh level hierarchy (eight quadrants). The X and Y coordinate measured in meters. Parameters setting illustrate in Table 1.

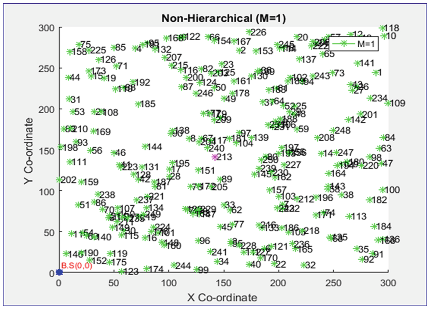


Fig. 3. 250 Nodes randomly deployed in 300 * 300 m^2

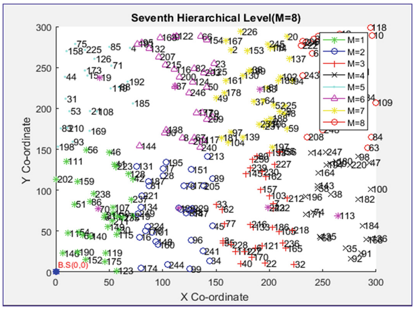


Fig. 4. 250 Nodes of seventh hierarchical level (eight quadrants) of 300 * 300 m^2

Table 1. Simulation and setting parameters

Parameters	Values
Network size	300 m *300 m
Number of nodes	250 nodes uniformly distributed
BS position	(0, 0)
Simulation rounds	400
Initial energy of node	0.2 J
Threshold energy	50 mJ
Energy for transmission	50 nJ per bit
Energy for reception	50 nJ per bit
Energy dissipation	100 pJ per bit
Data packet size	4000 bits

3.1 Results and Discussions

This section includes three experiments are used to illustrate and evaluate the performance of the proposed technique compared to other techniques like LEACH, PSO, and OHRT.

Experiment I analyzes the proposed algorithm performance in terms of a number of alive nodes at various hierarchy level and the residual energy which is estimated for 250 sensor nodes and 400 rounds in a network. Figure 5 shows that the number of alive nodes decreases with the increase in a number of rounds for variant levels of the hierarchy. It is also obvious that increasing hierarchy level balances energy consumption among clusters' members by increasing the number of CHs and delayed the nodes death, this leads in prolonging network lifetime.

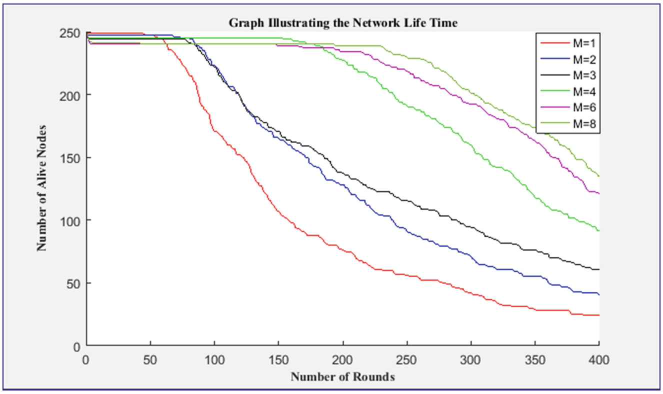


Fig. 5. Alive nodes number for different cluster no. (M)

Figures 6, 8, 10 and 12 show the residual energy in each node after 400 rounds for non-hierarchical (M = 1) and different level hierarchy (M = 2, 3, 8) respectively, while Figs. 7, 9, 11 and 13 show the histograms of the residual energy after 400 rounds of simulation for non-hierarchical (M = 1) and different level hierarchy (M = 2, 3, 8) respectively. It is obvious that the residual energy increases with the increase of the hierarchy level, this helps in extending the network lifetime. Increasing the clusters number could enhance the energy efficiency of the technique that presents a sign of a network lifetime enhancing.

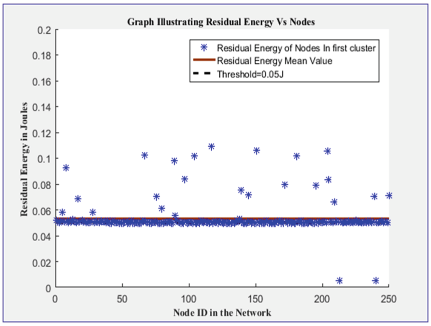


Fig. 6. Nodes' residual energy (J) (for M = 1)

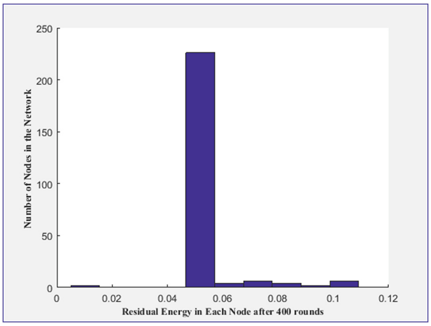


Fig. 7. Histogram of residual energy for (M = 1)

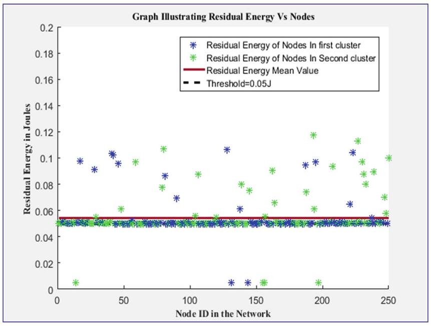


Fig. 8. Nodes' residual energy (J) (for M = 2)

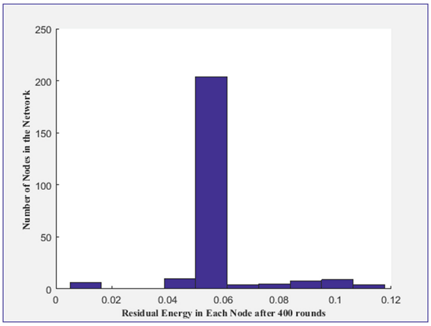


Fig. 9. Histogram of residual energy for (M = 2)

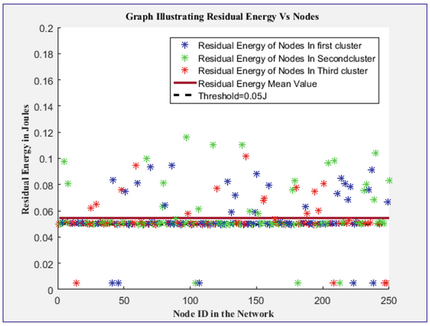


Fig. 10. Nodes' residual energy (J) (for M = 3)

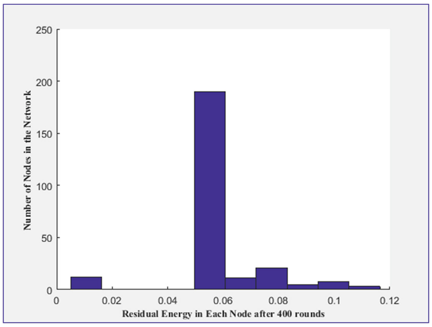
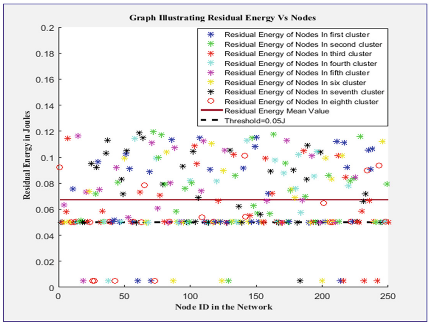


Fig. 11. Histogram of residual energy for (M = 3)



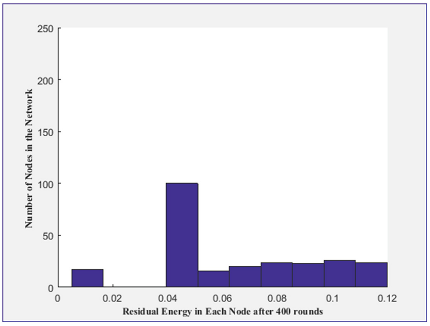


Fig. 12. Nodes' residual energy (J) (for M = 8)

Fig. 13. Histogram of residual energy for (M = 8)

Figure 14 shows the maximum value, mean value, standard deviation of the residual energy and number of live nodes after 400 rounds of 250 sensor nodes simulation for the proposed routing technique employed. Enhancing the residual energy mean value is achieved in each simulation round as the structure hierarchical expanded. This reflects preferable network performance as the nodes energy increase in the hierarchy latter level (M). It is clear that non-hierarchical technique has the smallest mean value and the eight-level hierarchy has the highest mean value.

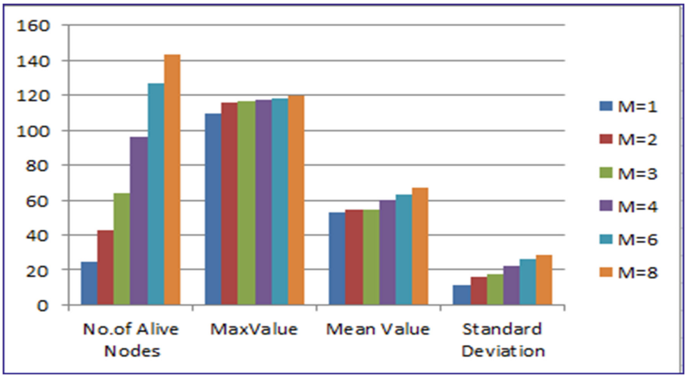
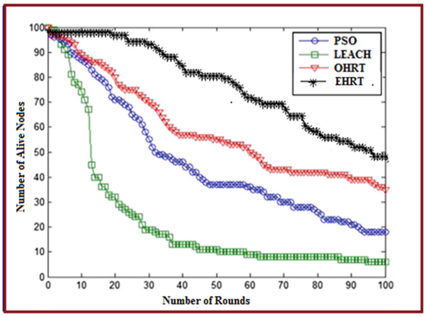


Fig. 14. Numbers of alive nodes, Max. value, Mean value and Standard deviation of the residual energy in (mJ) and after 400 rounds

Experiment II, Fig. 15 shows the number of alive nodes after each round for a number of nodes (100) and 100 rounds. The simulation results clearly show that after 100 rounds, the active nodes number in the proposed technique (EHRT) is higher than LEACH, PSO, and OHRT [10].

The network lifetime could be observed from the nodes death in the network. Figure 16 clearly displays the nodes death in both LEACH, PSO and OHTR, node death occurred earlier compared with the proposed technique. This in turns leads to extending the network lifespan.



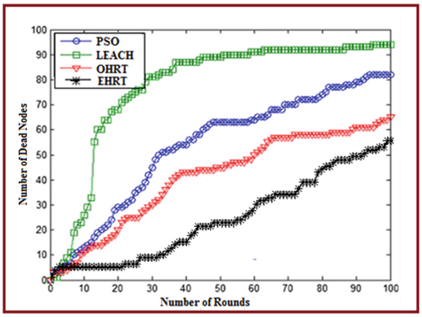


Fig. 15. Number of a live node in LEACH, PSO, OHRT and the proposed technique at different rounds

Fig. 16. Number of dead nodes in LEACH, PSO, OHRT and the proposed technique at different rounds

Experiment III resolves the performances of the four techniques. Figure 17 shows the network lifetime of the proposed, OHRT, PSO, and LEACH with a different number of nodes (for 500 nodes and 2000 rounds network). The proposed technique outperforms the other three techniques due to two reasons: First, Swapping cluster heads role can adjust energy consumption among cluster members. Second, it considers distance and nodes residual energies and elects optimum cluster heads for each cluster separately that can preserve more energy in nodes.

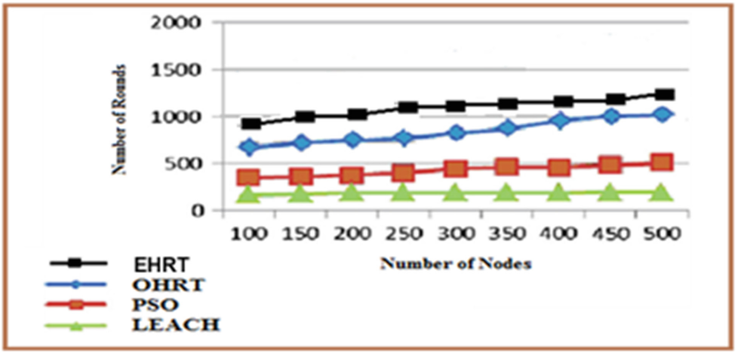


Fig. 17. Network lifetime

4 Conclusions and Future Work

Wireless Sensor Networks have different characteristics rather than traditional networks. Many constraints, such as computational power, storage capacity, and energy supply are considered while designing such type of networks. This paper presents an Enhanced Hierarchical Routing Technique (EHRT) which aims to reduce consuming energy and extend network lifetime.

EHRT mainly focuses on the clustering principle algorithm using LEACH routing protocol which optimized by using a modified AFSA that modifies the fitness function of AFSA. EHRT compromises between the CHs calculated using the following, praying and swarming behaviors and then selects the optimal CHs locations for each cluster separately. EHRT is evaluated and compared to LEACH, PSO, and OHRT using different environments. EHRT achieves best results comparing to other techniques in terms of energy consumption and network lifetime.

In the future, we aim to work in improving the efficiency of energy routing techniques. Modification of the proposed protocol to be applied in real life large area networks; heterogeneous and mobile network (base station and sensor nodes) will be considered. Try to increase the network lifetime and consume all nodes energy effectively by optimizing the number of hierarchy level. Also, decrease AFSA consumption time to be able to spread more behaviors at the beginning of each round.

References

1. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. Comput. Netw. **38**, 393–422 (2002)
2. Zheng, J., Jamalipour, A.: Wireless Sensor Networks: A Networking Perspective. Wiley, Hoboken (2009)
3. Iqbal, M., Naeem, M., Anpalagan, A., Ahmed, A., Azam, M.: Wireless sensor network optimization: multi-objective paradigm. Sensors **15**, 17572–17620 (2015)
4. Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks. In: 2000 Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, vol. 2, 10 p. (2000)
5. Manjeshwar, A., Agrawal, D.P.: TEEN: a routing protocol for enhanced efficiency in wireless sensor networks. In: null, p. 30189a (2001)
6. Sendra, S., Parra, L., Lloret, J., Khan, S.: Systems and algorithms for wireless sensor networks based on animal and natural behavior. Int. J. Distrib. Sens. Netw. **11**, 625972 (2015)
7. Guo, T., Zhao, H.: An Improvement of AFSA in global search with scout swarms. In: 2013 International Conference on Advanced Computer Science and Electronics Information (ICACSEI 2013) (2013)
8. Dhiman, V.: BIO inspired hybrid routing protocol for wireless sensor networks. Int. J. Adv. Res. Eng. Technol. **1**, 33–36 (2013)
9. Bhaduri, S.N., Fogarty, D.: New methods in ant colony optimization using multiple foraging approach to increase stability. Advanced Business Analytics, pp. 131–138. Springer, Singapore (2016).
10. El-Said, S.A., Osamaa, A., Hassanien, A.E.: Optimized hierarchical routing technique for wireless sensors networks. Soft Comput. **20**, 4549–4564 (2016)
11. Xing, B., Gao, W.-J.: Fish inspired algorithms. Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms. ISRL, vol. 62, pp. 139–155. Springer, Cham (2014).
12. Ganesan, T., Vasant, P., Elamvazuthi, I.: Advances in Metaheuristics: Applications in Engineering Systems. CRC Press, Boca Raton (2016)

Multi-filter Score-Level Fusion for Fingerprint Verification

Muhammad Atta Othman Ahmed[1](✉), Omar Reyad[2], Yasser AbdelSatar[3], and Nahla F. Omran[3]

[1] Department of Electrical and Electronic Engineering, University of Cagliari,
Piazza D'Armi, 09123 Cagliari, Italy
muhammad.ahmed@diee.unica.it

[2] Sohag University, Sohag, Egypt
ormak4@yahoo.com

[3] South Valley University, Qena, Egypt
Yasser.abdelsatar@sci.svu.edu.eg, nahlaafathy@yahoo.com

Abstract. Biometric systems are widely used in various applications of today's authentication technology. The unimodal systems suffer from various stumbling blocks such as noisy inputs, non-universality, intra-class variability and imposter spoofing which affects the system performance and accuracy. To effectively handle these problems, two or more individual modalities are used. In this paper, we presented a multimodal approach for fingerprint verification based on a combination of score level fusion rules. In the preprocessing stage, Anisotropic Diffusion Filter (ADF) and Histogram Equalization (Hist-Eq) techniques were applied to overcome the main challenging drawbacks of fingerprint samples acquisition such as distortion, noise, rotation, etc. Supplementary, the Local Binary Pattern (LBP) was used for feature extraction. In score level fusion, the matching scores of individual fingerprints were combined via several fusion rules. Receiver Operating Characteristics (ROC) curves were formed for the multimodal approach that's why it is mainly used to evaluate our system. Experimental results shown improvements of the multimodal system using ADF and Hist-Eq versus the unimodal non-preprocessed fingerprint samples. The obtained results indicated that there is a significant increase in the performance of the proposed system due to the combination of scores, making it suitable for more applications relevant to identity verification.

Keywords: Fingerprint verification · Anisotropic diffusion filter
Histogram equalization · Score-level fusion · ROC · DET ware
CMC curve

1 Introduction

Access to one's personal information might soon require only his body as the password by the means of biometric systems. Biometrics is a multi-disciplinary field concerned with measuring human physiological and behavioural traits.

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 624–633, 2018.
https://doi.org/10.1007/978-3-319-74690-6_61

Physiological characteristics such as face [1], fingerprint [2], DNA and Iris and behavioural traits include voice, signature and keystroke, all were used as an identity to recognize individuals. Among all biometric traits, fingerprints are the most hopeful and trustworthy individual recognition applications which have been positioned in a wide range of secure applications [3,4]. However, the raw images from most fingerprint sensor devices are affected by noise, distortion, and displacement between each two fingerprint images acquired from the same finger [5]. Fingerprint authorization and authentication are becoming a challenging task of the highest accuracy with the lowest probability of fake acceptance rates and reliable non-rejection rate. Any fingerprint recognition system can be addressed by a verification mode or an identification mode [6]. The former depends on matching the individual sample with his/her template stored in the database and determine weather it belong to the same identity or not. The latter objective a person biometric acquired is compared with all persons templates enrolled in a system database and the system determines either the highest degree of authentication with the person's input or a person presenting the identity is not enrolled. Because of noise can interfere with the extraction, false minutiae may be detected and true minutiae may be missed.

The aim of this paper is to present an effective approach for fingerprint verification via fusing matching scores to collect the most relevant features. The proposed method use a combination of two main enhancement techniques Anisotropic Diffusion Filter (ADF), Histogram equalization (Hist-Eq) considering different rules (Min, Max, Sum and the Dot product) at level scores which proved the highest performance results in the fingerprint identification algorithm. The algorithm is tested on low-quality images from the FVC2006 (DB2A).

The paper is organised as follows. In Sect. 2, presented the related works. In Sect. 3, the preliminaries of ADF, Hist-Eq and LBP is given. In Sect. 4, the fusion method at score level is proposed. The experimental results and analysis are discussed in Sect. 5 while conclusions are shown in Sect. 6.

2 Related Works

To overcome the drawbacks of unimodal biometric systems multimodal systems are being developed. In multimodal systems, the biometric fusion methods have created an increasing attention and interest in the researchers because of their high recognition rates. A new technique to fuse the image enhancement methods based on image filtering techniques is presented in [2] to enhance the poor quality of fingerprint images and increase the clarity of ridges and valleys of the image features which lead to expanding robust fingerprint features. In biometric systems, fusion level concerned with the accuracy of recognition systems which can be increased per the type of fusion: fusion at feature level [7], decision level, matching level [8] and score level fusion [9]. Feature level fusion based on fuse features for more than two biometric traits, the data are obtained from each sensor, then, compute a feature vector for each one and fuse the two feature vectors into a single new feature vector. Feature reduction methods are useful for a larger set of features. Feature level can cause a conflict or suffer from high dimensional. Decision fusion used to make the final decision between the multiple biometrics data results and classified it into two classes to determine acceptance or rejection. But the decision level fusion can occur possibility of errors because of the limited availability of information. Matching score level these techniques implemented to minimize the false rejection rate (FRR) for a given false acceptance rate (FAR). Therefore, the fusion at score level is preferred due to simple in accessing and integration of scores. In score level fusion, fusion can be performed in two distinct approaches namely classification approach and combined approach. With its efficiency and elasticity, fusion at score level becomes a superior fusion technique [10]. Finally, the match scores may follow diverse probability distributions, may provide fully different accuracy's and may be correlated. Convolutional neural networks guarantee better accuracy in most fingerprint images processing [11] and biometrics applications [12]. A multimodal biometric system of iris and fingerprint using feature level fusion is proposed in [13]. In feature level fusion, the processing time increased because of high dimensional feature sets. Score level fusion in turn will combine the matching scores of individual modalities.

3 Preliminaries

Low quality, noise, distortion, rotation are the main challenges against building a robust authentication biometric systems based on human fingerprints. Our methodology employes the ADF and Hist-Eq techniques in the preprocessing stage to overcome the mentioned challenges. Then, we used the well-known LBP algorithm for feature extraction process. The matching score fusion rules Min, Max, Sum and the Dot product are utilized to obtain the highest fingerprint verification system performance.

3.1 Anisotropic Diffusion Filter

Anisotropic non-linear diffusion filter is presented in [14] as a powerful image processing technique. It enables to simultaneously remove the noise and enhance sharp features in two and three dimensional images. ADF depends on a set of parameters which are crucial, conductance function, gradient threshold and the stopping parameter which determine the range and the conduct of the diffusion. The formula for ADF is given as:

$$\frac{\partial I\left(x,y,t\right)}{\partial t} = div\left[\|g(\bigtriangledown I(x,y,t)\|\, g(\bigtriangledown I(x,y,t))\right] \tag{1}$$

where $\mathrm{I(x,y,0)}$ is the original image, t is the time parameter, $\bigtriangledown I(x,y,t)$ is the gradient of the image at time t and $g()$ is the function of conductance.

3.2 Histogram Equalization

Hist-Eq technique is a common approach for adjusting image intensities in order to enhance the poor-quality image contrast. Hist-Eq increased the contrast of the image by lowering the number of gray levels in that image [15]. In the equalizing process, the neighbouring gray levels with light probabilistic density are fused into one gray level, while the gap between two neighbour gray levels with heavy probabilistic density is increased. Let f be a given image represented as a $[m_r{*}n_c]$ matrix of integer pixel intensities ranging from 0 to 255, L is the number of possible intensity values, often 256. Let P denote the normalized histogram of v with a bin for each possible intensity. So

$$p_n = \frac{\textit{Number of pixels with intensity } n}{\textit{Total number of pixels}}, \quad n = 0, 1, \ldots, L-1 \tag{2}$$

The histogram equalized image g will be defined by

$$g_{i,j} = \lfloor (L-1) \sum_{n=0}^{f_{i,j}} p_n \rfloor \tag{3}$$

where the floor $\lfloor . \rfloor$ rounds down to the nearest integer.

3.3 Local Binary Pattern

LBP is considered as an efficient method for texture operator. It is based on a local region around each pixel of the image by thresholding the neighbourhood of each pixel then extract the result as a binary value [8]. In addition, a uniform pattern is used for decreasing the length of the feature vector and do a simple rotation-invariant descriptor. When LBP labels computed, uniform patterns are used separated label for each uniform pattern and all the non-uniform patterns are labeled together. At first LBP operator is limited to a small 3 × 3 neighbourhood with limit features. The main rotation invariant LBP operator, indicated here as LBPriu2, is achieved by circularly rotating every bit pattern to the minimal value. To overcome this, the LBP expanded to use neighbourhoods of different sizes. Figure 1 show the extended LBP operator which has two parameters (P, R). P is several neighbours sampling points on a circle of radius of R. So, when using (8, R) neighbourhood, there are a total of 256 patterns, which yields in 59 different labels.

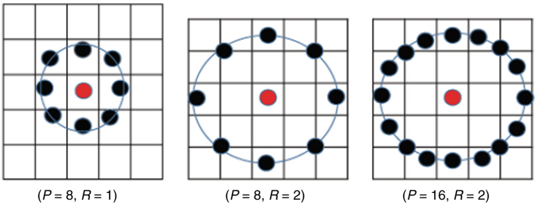


Fig. 1. A different sizes of LBP neighbourhoods. Figure source: M Sultana et al.; Local binary pattern variants-based adaptive texture features analysis for posed and nonposed facial expression recognition.

3.4 The Used Dataset

In this work, the Fingerprint Verification Competition 2006 [16] (FVC 2006) of fourth edition benchmark databases for a fingerprint verification software assessment is used. In FVC 2006, a fingerprint impressions were collected with four sensor devices about 150 fingers wide and 12 samples per fingertip in depth (1800 fingerprint images). Data acquired in FVC2006 was obtained with low quality index for most difficulties such as image distortion, huge rotation and displacement and wet/dry impressions.

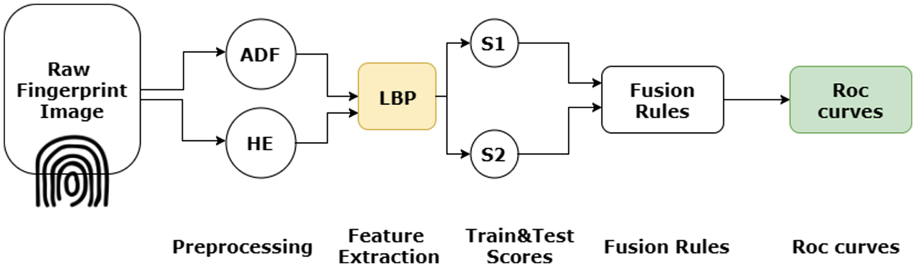


Fig. 2. Block diagram of the proposed fingerprint identification system.

4 The Proposed Identification Model

The main objective of our approach is overcoming the fingerprint verification stumbling blocks such as distortion, rotation and eliminating noise types while preserving the minutiae points and ridge structures. We proposed a new comprehensive method based on fusing the outcome of ADF and Hist-Eq techniques at score level. For a fingerprint identification system to increase the authentication accuracy, fusion is done at the score level. The proposed fusion rules listed as follows:

1. Minimum Score: $_s(S_1, S_2)$.
2. Maximum Score: $_s(S_1, S_2)$.
3. Summation of Scores: $\sum(S_1, S_2)$.
4. Dot Product Scores: $|S_1 \cdot S_2|$.

Where each score $S_i = \{s_1, s_2, \ldots, s_N\}, N = 59$. The matching scores between fingerprint templates and test samples aimed to be verified are calculated by the Euclidean distance:

$$S_{matching} = \sqrt{\sum_{i=1}^{n} (S_i - S_j)^2}. \tag{4}$$

Then, fused scores are evaluated to obtain the evidence of our approach relevance. Figure 2 shown the block diagram of the proposed approach.

5 Experimental Results

The proposed method is tested using different quality images from the fourth edition benchmark FVC database (FVC2006–DB2A). To assess the fingerprint identification system performance, we used Receiver Operating Characteristic curve (ROC curve), the Cumulative Match Curve (CMC) and Detection Error Tradeoff (DET) curves. Then, the obtained results are analysed and evaluated.

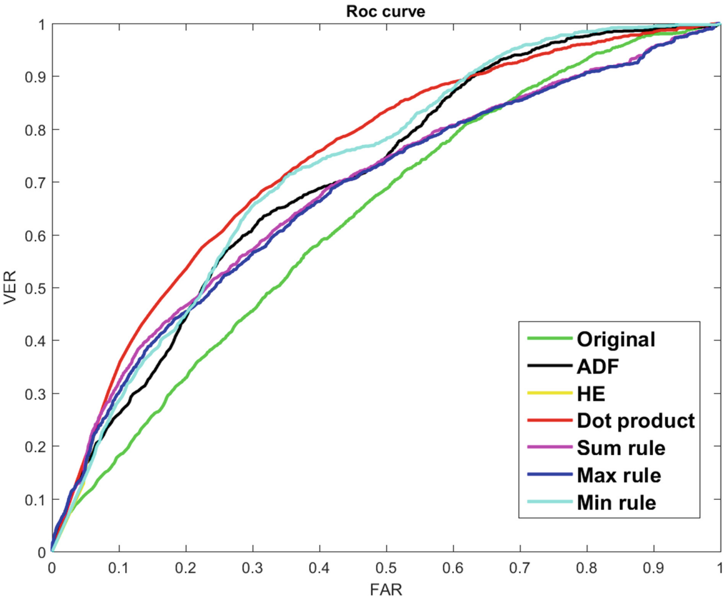


Fig. 3. ROC curves for fingerprint verification experiments ADF filter, Hist-Eq technique and different score fusion rules using (FVC2006-B2A) database show increase the verification rate by Dot product and Min score level fusion rules.

5.1 Receiver Operating Characteristics Curve

ROC curve is a graphical plot that illustrates the performance analysis and evaluation of biometric system as its discrimination threshold is varied [17]. The curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. In Fig. 3, the evaluation of ROC curves is shown for ADF, Hist-Eq and different fusion rules based on the scores for ADF filtering and Hist-Eq techniques at score level. It is found that the best fusion identification rate from the Dot product and Min score level fusion rules and the performance improvement is obtained. Thus, the results clearly explain that high identification rate could overcome fingerprint image distortion, noise and false minutiae drawback effects. The fingerprint verification rates using original fingerprints, preprocessing methods, and using proposed score-level fusion techniques at FAR = 0.001 and FAR = 0.01 are presented in Table 1. Results shown that there is remarkable effectiveness of the preprocessing methods, especially the Hist-Eq, compared to the use of original fingerprints. Also, the score level fusion rules outperformed the preprocessing individually. Meanwhile, the computational run-time are presented in Table 2.

5.2 Cumulative Matching Characteristic Curve

CMC curve is used to evaluate the ranking accuracy and it presents various probabilities of identifying an individual depending on how similar their features are to other individual's features in the database. Figure 4 shows the CMC curves obtained with the considered image enhancement techniques. The presented enhancement methods with four different fusion rules are compared as

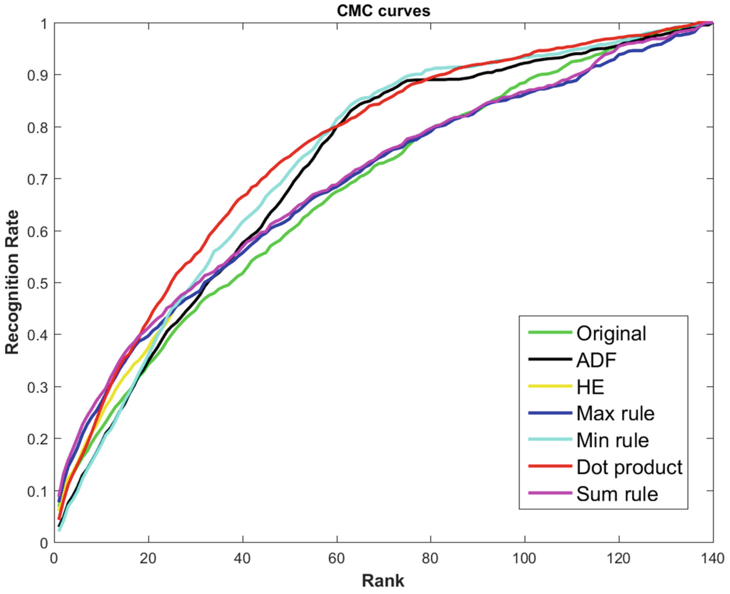


Fig. 4. CMC curves for fingerprint verification experiments ADF filter, Hist-Eq technique and different score fusion rules using (FVC2006-B2A) database.

Table 1. Fingerprint verification rates at FAR = 0.001 and FAR = 0.01 for the different preprocessing methods + LBP and fusion rules.

		PreP		Fusion rules			
	Original	ADF	Hist-Eq	$Min_{S_1}^{S_2}$	$Max_{S_1}^{S_2}$	$\sum(S_1, S_2)$	$S_1 \cdot S_2$
VER@FAR = 0.001	0.0045	0.0072	0.1775	**0.1879**	0.0093	0.0877	**0.189**
VER@FAR = 0.01	0.0135	0.1304	**0.3937**	**0.3995**	0.2021	0.1996	**0.4132**

Table 2. Computational run-times of the pre-processing filters and LBP feature extraction (in second). Experiments performed on PC, Intel core i5, macOS Sierra, 8 GB ram using MATLAB R2017b; UNICA.it Academic Licence.

	PreP		Fusion rules			
Original	ADF	Hist-Eq	$Min_{S_1}^{S_2}$	$Max_{S_1}^{S_2}$	$\sum(S_1, S_2)$	$S_1 \cdot S_2$
0.037615	3.470380	**0.688505**	**0.9015**	0.9328	0.9609	**0.9107**

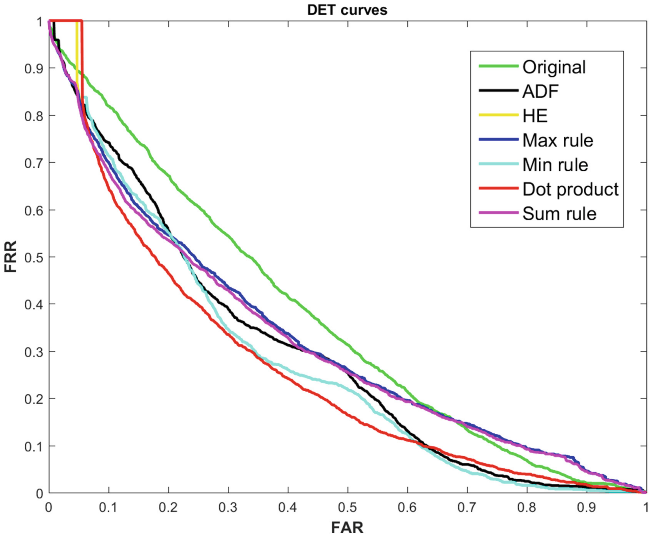


Fig. 5. DET ware curves for fingerprint verification experiments ADF filter, Hist-Eq technique and different score fusion rules using (FVC2006-B2A) database show decreasing error rate by dot product and min score level fusion rules.

shown in Fig. 4. It is found that Dot product and Min fusion rules at score level have high recognition rates which lead to image enhancement increasing.

5.3 Detection Error Tradeoff Ware Curve

DET ware curve is a graphical plot of error rates for biometric systems evaluation, plotting the false non-matching rate vs. false match rate. In Fig. 5, the performance of ADF and Hist-Eq technique using DET curves is shown. Results show that ADF filtering has a lower error rate versus Hist-Eq technique. The results reported in that Dot product and Min score fusion rules outperformed the rest of configurations.

6 Discussion and Conclusion

Unimodal fingerprint verification systems proven to be improved using multimodal systems, matching score combination rules is used for this purpose. In this paper, we proposed a multimodal approach rely on the fusion of matching score. ADF and Hist-Eq are applied in the preprocessing stage to overcome the main drawbacks in fingerprint acquisition systems. Then, feature extraction is done using the LBP algorithm. Several matching score fusion rules such as Min, Max, Sum and Dot product are applied and evaluated by means of verification accuracy computational complexity and run-time. The performance of Dot product and Min score fusion rules proven to outperformed other considered fusion rules, we recommend further investigation for their usefulness in this state of art.

References

1. El-Sayed, M.A., Khafagy, M.A.: An identification system using eye detection based on wavelets and neural networks. arXiv preprint arXiv:1401.5108 (2014)
2. Khfagy, M., AbdelSatar, Y., Reyad, O., Omran, N.: An integrated smoothing method for fingerprint recognition enhancement. In: International Conference on Advanced Intelligent Systems and Informatics, pp. 407–416. Springer (2016)
3. Biggio, B., Fumera, G., Russu, P., Didaci, L., Roli, F.: Adversarial biometric recognition: a review on biometric system security from the adversarial machine-learning perspective. IEEE Sig. Process. Mag. **32**(5), 31–41 (2015)
4. Biggio, B., Fumera, G., Russu, P., Didaci, L., Roli, F.: Poisoning adaptive biometric systems. In: Structural, Syntactic, and Statistical Pattern Recognition, pp. 417–425. Springer, Heidelberg (2012)
5. Li, S.Z., Jain, A.: Encyclopedia of Biometrics, 2nd edn. Springer, New York (2015)
6. Rajeswari, P., Viswanadha Raju, S., Ashour, A.S, Dey, N.: Multi-fingerprint unimodel-based biometric authentication supporting cloud computing. In: Intelligent Techniques in Signal Processing for Multimedia Security, pp. 469–485. Springer, Cham (2017)
7. Jeng, R.-H., Chen, W.-S.: Two feature-level fusion methods with feature scaling and hashing for multimodal biometrics. IETE Tech. Rev. **34**(1), 91–101 (2017)
8. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern Anal. Mach. Intell. **24**(7), 971–987 (2002)
9. Castrillón-Santana, M., Lorenzo-Navarro, J., Ramón-Balmaseda, E.: Multi-scale score level fusion of local descriptors for gender classification in the wild. Multimedia Tools and Appl. **76**(4), 4695–4711 (2017)
10. Parkavi, R., Babu, K.R.C., Kumar, J.A.: Multimodal biometrics for user authentication. In: 2017 11th International Conference on Intelligent Systems and Control (ISCO), pp. 501–505. IEEE (2017)
11. El-Sayed, M.A., Estaitia, Y.A., Khafagy, M.A.: Automated edge detection using convolutional neural network. Int. J. Adv. Comput. Sci. Appl. **4**(10), 10–20 (2013)
12. Peralta, D., Triguero, I., García, S., Saeys, Y., Benitez, J.M., Herrera, F.: On the use of convolutional neural networks for robust classification of multiple fingerprint captures. Int. J. Intell. Syst. **33**(1), 213–230 (2018)

13. Anitha, T.N., Ravi, J., Geetha, K.S., Raja, K.B.: Bimodal biometric system using multiple transformation features of fingerprint and iris. Int. J. Inf. Technol. (ACEEE) **1**(3), 20 (2011)
14. Gerig, G., Kubler, O., Kikinis, R., Jolesz, F.A.: Nonlinear anisotropic filtering of MRI data. IEEE Trans. Med. Imaging **11**(2), 221–232 (1992)
15. Weickert, J.: Multiscale texture enhancement. In: Computer Analysis of Images and Patterns, pp. 230–237. Springer (1995). https://doi.org/10.1007/978-3-642-40246-3
16. Fingerprint verification Competition. WWW document (2006)
17. Gorodnichy, D.O.: Evolution and evaluation of biometric systems. In: IEEE Symposium on Computational Intelligence for Security and Defense Applications, CISDA 2009, pp. 1–8. IEEE (2009)

Color Image Segmentation of Fishes with Complex Background in Water

Ahmed M. Abdeldaim[3,4](✉),
Essam H. Houssein[2,4], and Aboul Ella Hassanien[1,4]

[1] Faculty of Computers and Information, Cairo University, Giza, Egypt
[2] Faculty of Computers and Information, Minia University, Minya, Egypt
[3] Culture and Science City, 6th October, Egypt
a7medabdeldaim@gmail.com
[4] Scientific Research Group in Egypt (SRGE), Giza, Egypt
srge1964@gmail.com
http://www.egyptscience.net

Abstract. Color image segmentation of fishes with complex background in water considered a big challenge. In this paper five segmentation methods for fishes are discussed; they are Grabcut algorithm, Otsu thresholding method, Edge detection technique, Mean-shift method and Region-Growing algorithm. However, most of them are manually segmentation methods or require a white or uniform background. In order to evaluate the segmentation methods, they were tested using a new dataset which contains about 270 fish species from natural scenes. The results revealed that the Grabcut Algorithm has achieved a very good results comparing with other methods.

Keywords: Multilevel Thresholding · Image segmentation
Fish recognition · Otsu · Grabcut

1 Introduction

Segmentation is an important part in image processing; it is the division of an image into regions or categories with similar attributes which is useful for image analysis and interpretation. Segmentation has many techniques such as: Edge detection [1] which attempts to capture the significant properties of objects in the image, Fuzzy c-means (FCM) clustering [2] which classifies an image by grouping similar data points in the feature space into clusters, Multilevel Thresholding [3] which segments a gray-level image into several distinct regions, and there are a plenty of techniques. Segmentation is used in many applications in many fields; it can be used in identification and detection of population, objects or animals.

Fish are a class of aquatic vertebrates [4]; they are different from all other animals as their gills and fins, also they spend all of their lives in the water and most of them are cold-blooded. Scientists believe that there are more than 24,000 different species of fish in the world. They range in size from the largest, Whale

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 634–643, 2018.
https://doi.org/10.1007/978-3-319-74690-6_62

shark at 16 m (51 ft) long, to the smallest the 8 mm (1/4 in.) stout infantfish. Using taxonomy ontology animals could be classified into hierarchical categories in a scientific methodology. Taxon is the basement of taxonomy, each taxon in the taxonomic tree has a top-to-bottom description to identify its hierarchical information which contains several concepts, known as Kingdom, Phylum, Class, Order, Family, Genus and Species.

Unfortunately there are not many automatic fish segmentation methods, most of them are partially manual. In this paper, we presented five fish segmentation methods; such as the Grabcut algorithm [5], Otsu thresholding method [6], Edge detection technique, Mean-shift method and Region-Growing algorithm. Moreover, all the presented segmentation methods are evaluated on dataset which contains about 270 fish species from natural scenes.

The remainder of this paper is organized as follows. Section 2 provides the related work. The Materials and methods used in this paper are presented in Sect. 3. Section 4, introduces the experimental results. Finally, Sect. 5 concludes the paper and suggests some directions for future studies.

2 Related Work

In this section, we will introduce briefly studies that related to our work. There are plenty of methods that depend on segmentation using Grabcut algorithm, Hernández et al., have used Grabcut to propose a full automatic Spatio-Temporal human segmentation methodology [7], they used a HOG-based person detector, face detection, and skin color model to initialize Grabcut seeds. Prakash et al., proposed a novel formulation for integrating Grabcut with Active-contour [8] to obtain an automatic foreground object segmentation, they depended on that the Active Contour cannot remove the holes in the interior part of the object. On other hand, Grabcut produces poor segmentation results in cases when the color distribution of some part of the foreground object is similar to background. So they proposed a segmentation technique, Snakecut, based on a probabilistic framework that provides an automatic way of object segmentation. Parkhi et al., segmented the foreground (pets) and background by Grabcut [9] this was done by using cues from the over-segmentation of an image (super pixels).

Chuang et al., proposed an automatic segmentation algorithm for fish sampled by a trawl-based underwater camera system [10], they achieved a 78% recall against the ground truth on the successful segmentation of fish, under very low-contrast underwater images. Li et al., presented a method to identify fish spices [11], they used basic image processing techniques to segment fish from background, and the used dataset was four fish respectively of chub, crucian, bream fish and carp, true color images obtained by digital camera. Takeshi Saitoh et al., introduced a fish image recognition method using feature points for fish images with complicated backgrounds [12], the feature points are four points: mouth, dorsal fin, caudal fin and anal fin. Each of these points is manually provided by the user and is designed as characteristic locations to avoid incorrect input by users. Storbeck et al., proposed a classification system for underwater video analysis [13], they defined a new method to recognize a large variety of underwater species by using a combination of affine invariant texture and shape features. Hu et al., presented a novel method of classifying species of fish based on color and texture features using a multi-class support vector machine (MSVM) [14].

3 Materials and Methods

3.1 Dataset Description

The used dataset was collected from http://fishesofaustralia.net.au/, it contains 270 images each image represent a different species. The dataset includes fish images from 2 classes, 7 orders, 25 families and 98 genus, Fig. 1, shows hierarchical classification of fish and Fig. 2, shows samples from the used dataset.

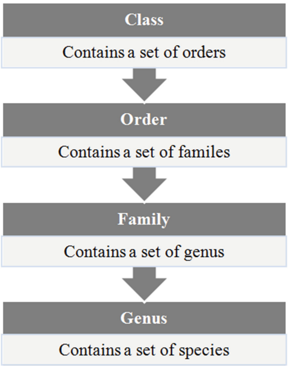


Fig. 1. Hierarchical classification of fish.

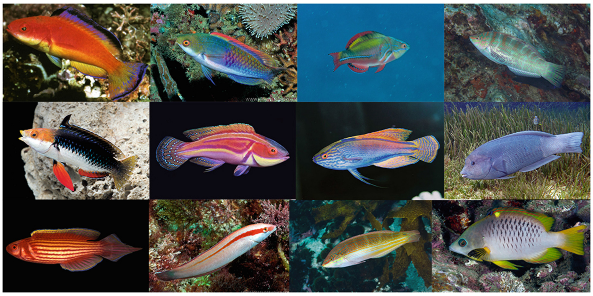


Fig. 2. Samples from the dataset.

3.2 Segmentation Methods

In this section, five segmentation methods are introduced and these methods were performed individually.

Segmentation Using Grabcut Algorithm. In this section Grabcut algorithm was used to segment fish from background. Grabcut is a foreground extraction algorithm that can be used when foreground and background color distributions are not well separated. It is based on graph cuts and works by specifying a bounding box around the object to be segmented, in our case we use the whole image as a bounding box, the algorithm estimates the color distribution of the target object and that of the background using a Gaussian mixture model [15]. To minimize the process time and optimize the algorithm quality the input images were resized to 256 * 256, Fig. 3, shows input images after resizing.

Grabcut algorithm is applied to images in RGB (Red, Green and Blue) color space and it is applied to images 4 times, each time images are flipped horizontally, vertically and horizontally-vertically then taking the intersection between these 4 images, Fig. 4, shows Grabcut algorithm results. To remove unwanted shapes from images, some of morphological operators were used such as: opening which erodes away the boundaries of foreground object, it is useful for removing small white noises, and Closing which useful in closing small holes inside the foreground objects, or small black points on the object.

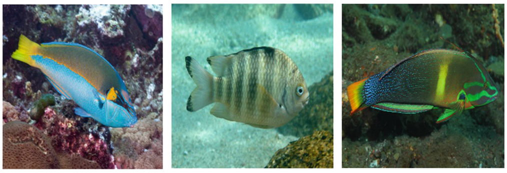


Fig. 3. Input images after resizing.

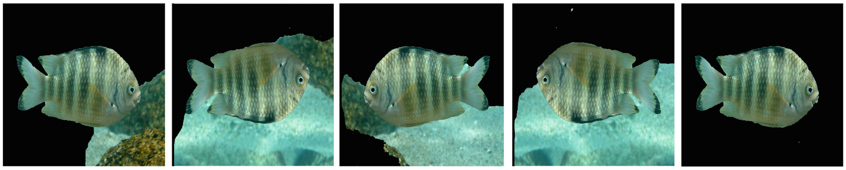


Fig. 4. From left to right: performing Grabcut on the original image, after flipping the image vertically, after flipping horizontally, after flipping vertically-horizontally and the intersection between the 4 images.

Segmentation Using Otsu Thresholding Method. Otsu thresholding method is used to convert a gray image to binary image, it assumes that the image contains two classes of pixels (foreground pixels and background pixels), it calculates the optimum value which separating the image foreground from background from a bi-modal histogram. Before applying Otsu, images were converted to HSV color space which stands for Hue, Saturation and Value. Because fish in all images are obvious in value component which was used in thresholding as the input image; Fig. 5, shows images in value component.

Before applying Otsu thresholding a blur operations was performed for the input image, then the histogram of the blurred image was calculated, the followed step was normalize the calculated histogram using Eq. 1, then the cumulative sum was calculated for the normalized histogram values, using the previous values, Mean is calculated from Eq. 2, and Variance is calculated from Eq. 3, then the threshold level calculated by multiplying the variance value and the cumulative value. Figure 6, shows the calculation of the threshold value; Fig. 7, shows images after applying Otsu thresholding.

$$NormalizedHistogram = \frac{H}{Max(H)} \tag{1}$$

$$M = \frac{(W * I)}{C} \tag{2}$$

$$Var = I * [W - M]^2 \tag{3}$$

Where H represents the calculated histogram values, W for the histogram weight, I for intensity values, C for cumulative values and M for the mean. After applying Otsu thresholding method, some of the above mentioned morphological operators were used in addition to boundary removal operator which removes any component touches the image boundaries; also min-area operator was used to remove small shapes in the image.

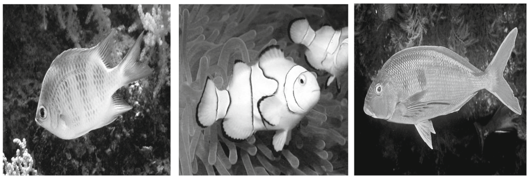


Fig. 5. Images in value component.

Segmentation Using Edge Detection. Edge detection is an image processing technique; it is used to find objects boundaries inside the image by detecting discontinuities in brightness. To achieve high results, images first were converted to HSV color space and the V component of HSV was used as an input image for edge detection. After applying edge detection, some of morphological operators were used like boundary removal operator which removes any component touches the image boundaries; also min-area operator was used to remove small shapes in the image. To overcome the uncompleted boundary paths problem, we used a boundary closing algorithm; Fig. 8 shows the result of applying boundary closing algorithm.

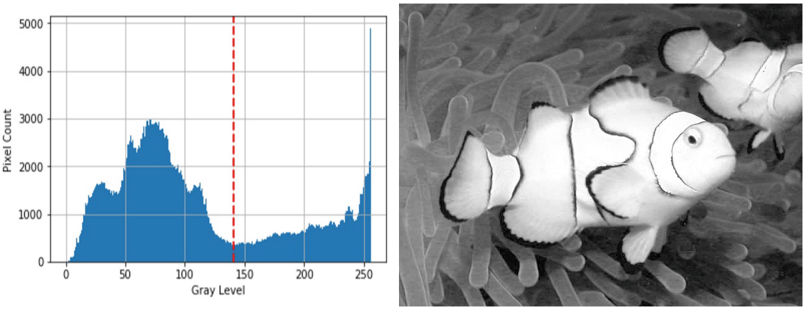


Fig. 6. The calculation of the threshold value, the red line in the histogram represents the threshold value.

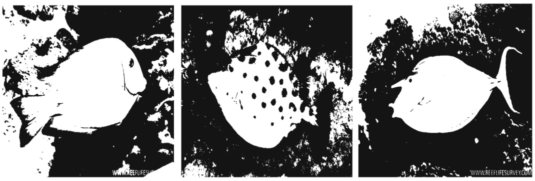


Fig. 7. Images after Otsu thresholding.

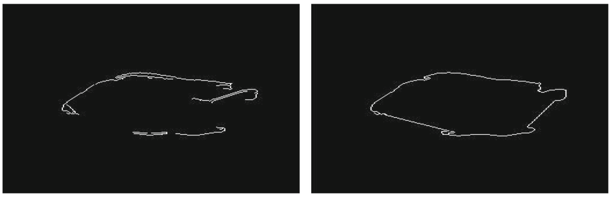


Fig. 8. The result after applying boundary closing algorithm.

Segmentation Using Mean-Shift. Mean shift is the most powerful clustering technique which is very useful for damping shading or tonality differences in localized objects; it is used in many fields such as image segmentation, clustering, visual tracking and space analysis. Because fish are more contrasted in the Value component, images were converted to HSV color space. Then the mean shift algorithm is applied, Edge detection is applied to extract fish from the background.

Segmentation Using Region Growing. In this section, Region Growing method was used. In general, Region-based methods compare one pixel with its neighbors. If a similarity criterion is satisfied, the pixel can be set belong to the cluster as one or more of its neighbors. In order to achieve high results, the center coordinates of the fish were entered manually by the user and then the region growing algorithm returns the segmented fish body.

4 Results and Discussions

This section presents the results for the five segmentation methods used in this paper such as Grabcut algorithm, Otsu thresholding method, Edge detection technique, Mean-shift method and Region-Growing algorithm. Figures 9, 10, 11 and 12, show the results from theses methods.

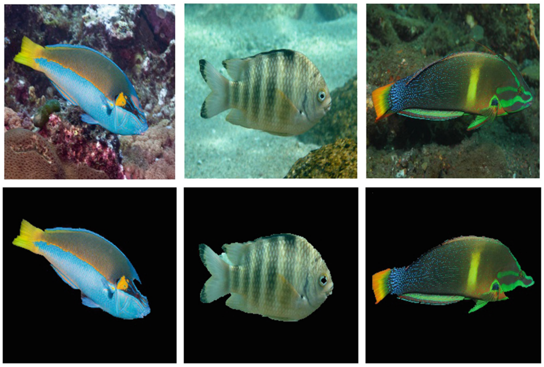


Fig. 9. First row represents images before segmentation; second row represents images after segmentation using Grabcut.

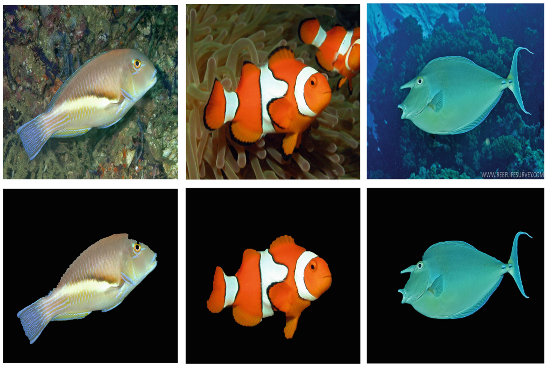


Fig. 10. First row represents images before segmentation; second row represents images after segmentation using Otsu thresholding.

In order to evaluate the segmentation methods three evaluation criteria such as RMSE, PSNR and SSIM [16] are utilized. RMSE is the root mean square deviation which is used as a measure of the difference between values, SSIM is the structural similarity index method which is used for measuring the similarity between two images and PSNR is the peak signal to noise ratio which is used

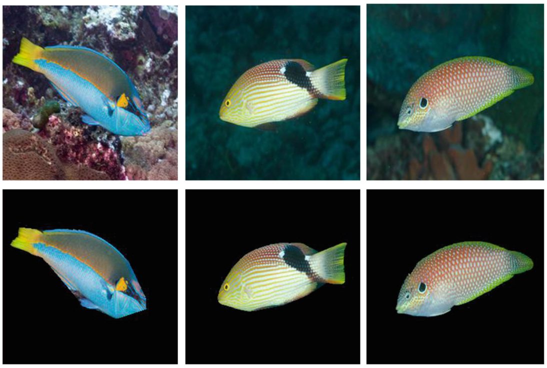


Fig. 11. First row represents images before segmentation; second row represents images after segmentation using Edge detection.

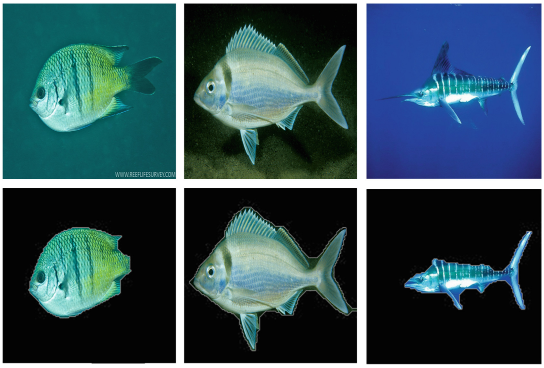


Fig. 12. First row represents images before segmentation; second row represents images after segmentation using Mean-shift.

to measure the quality of reconstruction of lossy compression codecs. RMSE is calculated from Eq. 4, SSIM is calculated from Eq. 5, and PSNR is calculated from Eq. 6. Table 1, shows the comparison results for the all segmentation methods. Generally, Table 1 shows that the Grabcut algorithm outperforms in terms of RMSE and PNSR over the compared segmentation methods.

$$RMSE = \sqrt{sum_{i=1}^{M} sum_{j=1}^{Q} (Org(i,g) - Seg(i,j))^2} \tag{4}$$

Where Org is the original image and the segmented image is Seg.

$$PSNR = 20 * \log_{10} \frac{255}{RMSE} \tag{5}$$

$$SSIM(Org, Seg) = \frac{(2\mu_{Org}\mu_{Seg} + C_1)(2\sigma_{Org,Seg} + C_2)}{(\mu_{Org}^2 + \mu_{Seg}^2 + C_1)(\sigma_{Org}^2 + \sigma_{Seg}^2 + C_2)} \tag{6}$$

Table 1. Comparison between all segmentation methods in terms of RMSE, PSNR and SSIM.

Criteria	Grabcut	Otsu	Edge detection	Mean-Shift	Region growing
RMSE	8.28	9.09	13.46	13.98	13.57
PSNR	1.52	1.47	9.58	9.23	9.29
SSIM	0.24	0.12	0.2	0.12	0.19

Where μ_{Org} and μ_{Seg} are the images mean intensity of the original and the segmented images, σ^2_{Org} and σ^2_{Seg} indicates the standard deviation of both images, $\sigma_{Org,Seg}$ represents the covariance of the both images and $C_1 = 6.5025$ and $C_2 = 58.52252$ as constants.

5 Conclusion and Future Work

In this paper, five segmentation methods are presented to detect and segment fishes from natural images even with different circumstances. To verify the evaluation of the theses method, the segmentation of a set of images was performed. The tests have been done on synthetic and real images (Fish dataset). These images have been chosen to test the ability of all five methods to segment fish which is difficult to discern, in presence of noise with any number of classes. The experimental results showed that the segmentation quality obtained by the Grabcut algorithm is satisfactory and better than the other methods. Also, It may be noted that the computation time of the Grabcut algorithm is independent of the size of the image and the number of iterations. It achieved the best results in segmentation. For future studies, automatic fish classification by color, texture is still required to be studied in the future. Also, it is worth to investigate the fish species classification by color, texture based on machine learning and meta-heuristic optimization algorithms.

References

1. Ziou, D., Tabbone, S., et al.: Edge detection techniques-an overview. Pattern Recognition and Image Analysis C/C of Raspoznavaniye Obrazov I Analiz Izobrazhenii **8**, 537–559 (1998)
2. Chuang, K.-S., Tzeng, H.-L., Chen, S., Wu, J., Chen, T.-J.: Fuzzy c-means clustering with spatial information for image segmentation. Comput. Med. Imaging Graph. **30**(1), 9–15 (2006)
3. Arora, S., Acharya, J., Verma, A., Panigrahi, P.K.: Multilevel thresholding for image segmentation through a fast statistical recursive algorithm. Pattern Recogn. Lett. **29**(2), 119–125 (2008)
4. Webb, P.: Body form, locomotion and foraging in aquatic vertebrates. Am. Zool. **24**(1), 107–120 (1984)
5. Rother, C., Kolmogorov, V., Blake, A.: Grabcut: interactive foreground extraction using iterated graph cuts. ACM Trans. Graph. (TOG) **23**(3), 309–314 (2004)

6. Liu, D., Yu, J.: Otsu method and k-means. In: Ninth International Conference on Hybrid Intelligent Systems, HIS 2009, vol. 1, pp. 344–349. IEEE (2009)
7. Hernández, A., Reyes, M., Escalera, S., Radeva, P.: Spatio-temporal grabcut human segmentation for face and pose recovery. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 33–40. IEEE (2010)
8. Prakash, S., Abhilash, R., Das, S.: Snakecut: an integrated approach based on active contour and grabcut for automatic foreground object segmentation. ELCVIA: Electron. Lett. Comput. Vis. Image Anal. **6**(3), 13–28 (2007)
9. Parkhi, O.M., Vedaldi, A., Zisserman, A., Jawahar, C.: Cats and dogs. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3498–3505. IEEE (2012)
10. Chuang, M.-C., Hwang, J.-N., Williams, K., Towler, R.: Automatic fish segmentation via double local thresholding for trawl-based underwater camera systems. In: 2011 18th IEEE International Conference on Image Processing (ICIP), pp. 3145–3148. IEEE (2011)
11. Li, L., Hong, J.: Identification of fish species based on image processing and statistical analysis research. In: 2014 IEEE International Conference on Mechatronics and Automation (ICMA), pp. 1155–1160. IEEE (2014)
12. Miyazono, T., Saitoh, T., Shibata, T.: Feature points based fish image recognition. Int. J. Comput. Inf. Syst. Ind. Manag. Appl. **8**, 12–22 (2016)
13. Storbeck, F., Daan, B.: Fish species recognition using computer vision and a neural network. Fish. Res. **51**(1), 11–15 (2001)
14. Hu, J., Li, D., Duan, Q., Han, Y., Chen, G., Si, X.: Fish species classification by color, texture and multi-class support vector machine using computer vision. Comput. Electron. Agric. **88**, 133–140 (2012)
15. Ju, Z., Liu, H.: Fuzzy gaussian mixture models. Pattern Recogn. **45**(3), 1146–1158 (2012)
16. Hore, A., Ziou, D.: Image quality metrics: PSNR vs. SSIM. In: 2010 20th International Conference on Pattern Recognition (ICPR), pp. 2366–2369. IEEE (2010)

Analysis of Credit Risk Prediction Using ARSkNN

Ashish Kumar[1](✉), Roheet Bhatnagar[1], and Sumit Srivastava[2]

[1] Department of Computer Science and Engineering,
Manipal University Jaipur, Jaipur, Rajasthan, India
aishshub@gmail.com, roheet.bhatnagar@jaipur.manipal.edu

[2] Department of Information Technology,
Manipal University Jaipur, Jaipur, Rajasthan, India
sumit.srivastava@jaipur.manipal.edu

Abstract. Credit risk is characterized as the risk that borrowers will neglect to pay its advance commitments and loan obligations. It is very hard to predict the outcomes (risky borrower) manually as the evaluation of large features set is quite time consuming. That's why, we need some good predictor as classifier. The traditional k-NN is one pre-established classifier used in various domains along with credit risk predictions. The newly conceptualized ARSkNN is another such classification which reduces the runtime in predicting the outcomes and improves overall accuracy percentage of the predicted classes over Traditional k-NN. The method adopt the similarity measure which is based on the Mass estimation rather than distance estimation for predicting the K- nearest neighbor. The results were compared using WEKA 3.7.10 as tool and found significant improvement vis-á-vis the evaluation parameters by the ARSkNN method.

Keywords: Classification · Nearest neighbors · ARSkNN · Credit risk

1 Introduction

Credit will happen as a result of plentiful reasons: house loans or bank mortgages, automobile purchase, credit card purchases, and so on. Basel Committee on Banking Supervision defined credit risk as the potential of a counterparty or bank debtor will be unsuccessful to pay its debts in accord with pre-established terms [1]. This credit risk analysis is vital to monetary establishments which offer loans to businesses and people. In recent years, Indian banks have seen a massive increase in their credit card customers. According to the latest available established data from the Reserve Bank of India (RBI), the growing rate of credit cards in India is nearly 24%. Also till March 2016, more than twenty four million credit cards are issued to their customers by all banks in India. This increases the credit card loan risk of being defaulted. Credit provider banks regularly collect immense volume of data about borrowers to fathom risk levels of credit borrowers. This data

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 644–652, 2018.
https://doi.org/10.1007/978-3-319-74690-6_63

has additionally been utilized with analytical predictive methods to evaluate or to determine risky and unsafe clients associated in credits and loans.

The possibility that a credit card aspirant will default must be estimated from information about the aspirant provided at the time of the application, and the estimate will assist as the basis for accepting or rejecting his application. While classifying, monetary background and subjective aspects of credit borrowers are assessed. Among these, monetary ratios perform an vital role for risk level estimation [2]. The implementation of Basel Committee's principle turns out to be a daily decision based on a binary classification problem distinguishing good payers from bad payers [3]. The first researches on credit scoring were done by [4,5] who applied linear and quadratic discriminant analysis respectively to categorize credit applications as "good" or "bad" ones. Since precise classification is of advantage both to the creditor and to the aspirant, many statistical methods, including multivariate discriminant analysis [6], logistic regression [7], and nearest neighbor [8], have been used to develop models of risk prediction. With the evolution of artificial intelligence and machine learning, artificial neural networks [3,9,10] and classification trees [11], were also employed to forecast credit risk. According to [12], "Despite the intense study of credit scoring, there is no consensus on the most appropriate classification technique to use."

This paper tackles the following question: How to predict good and bad borrowers more accurately so that banks can reduce credit risk? At first the authors explore the traditional nearest neighbor techniques to predict the defaulter and then further establishes ARSkNN a new nearest neighbor classification technique, for analyzing the credit risk.

The paper is divided into different sections and in the following sections, authors review the pre-established kNN classifier models in the domain of credit risk. Section 2 provides an introduction to credit risk. Nearest Neighbor classifier and ARSkNN are described in Sects. 3 and 4 respectively. Evaluation Parameters used to evaluate these classifiers are discussed in Sect. 5. Description of the dataset used is mentioned in Sect. 6. Results and Discussions forms Sect. 7. Conclusion and Future Work is given in Sect. 8.

2 Credit Risk - An Introduction

Banks should target on three kinds of risk: Credit, Operational and Market [13]. Among these, credit risk is one of the biggest risk faced by most the banks. Credit Risk can be defined as a loss in sense of money, a bank would undergo, if a bank's debtor is unsuccessful to fulfill his commitments viz; partially or fully pay interest money on borrowed loan, partially or fully refund the amount borrowed in line with the concurred terms and conditions [14].

Credit Risks are premeditated based on the debtors' complete capability to pay back. To evaluate credit risk on a customer loan, creditors investigate the five C's namely: the candidate's **credit history**, his refund **capability**, his **capital**, the **credit's conditions** and related **collateral**.

Credit Risk is frequently characterized by three factors: loss risk, default risk, and exposure risk. Default and credit risk are generally synonymous. Credit

Risk management is a technique involving following steps – recognition of possible risks, the assessment of these risks, the relevant treatment, and at last the employment of risk models [15].

Assessment of Credit Risk is an important activity to avoid immense amount of losses for any financial institution. The financial institution follows a robust framework to successfully diminish and anticipate credit risks [16]. Thus in a nutshell, the comprehensive objective of credit risk assessment is to equate the features of a going to be debtor with other previous debtors, whose loans they have already deposited back.

3 Nearest Neighbor Classifier

The classification methods can be broadly classified into parametric and nonparametric problems. In fact, parametric methods are based upon the assumptions of normally distributed population and estimate the parameters of the distributions to solve the problem [17]. However, according to Berry and Linoff [18] nonparametric methods make no assumptions about the specific distributions involved, and are therefore distribution-free. The k-nearest neighbor classifier serves as an illustration of a non-parametric statistical approach.

The cornerstones of k-nearest neighbor classification [19] are Nearest Neighbor (NN) classifier and the k-NN rule proposed by Fix and Hodges in 1951. It is also acknowledged by names such as instance based classification, memory based classification, case based classification and much more. There are three key building blocks of a k-NN classifier: a set of class labeled instances; a dissimilarity (e.g. Euclidean Distance) or similarity metric to work out distance or similarity among two instances; and the value of k i.e., the number of nearest neighbors to be considered.

According to Berry and Linoff [18] "the choice of k also affects the performance of the k-NN algorithm. This can be determined experimentally. Starting with $k = 1$, we use a test case to estimate the error rate of the classifier. This process is repeated each time by incrementing k to allow for one more neighbors. The K-value that gives the minimum error rate may be selected. In general, larger the number of training samples is, the larger the value of k will be." Various metrics have been suggested to enhance the k-NN classifiers for example, Mahalanobis distance [20], adaptive distance [21] and local metric [22]. Moreover K-NN classifier requires an equal number of good and bad sample cases for better performance [8].

4 ARSkNN - A Novel Mass Based Classifier

ARSkNN, which is conceptualized by the same authors [23], is an efficient k-nearest neighbor classifier exploits Massim, a mass-based similarity measure in spite of utilizing any distance-based similarity measures.

ARSkNN has got two stages: 1. Modeling Stage and 2. Class Assignment Stage. In modeling (preprocessing) stage, a Similarity Forest (sForest) with t number of Similarity Trees (sTrees), is built from D dataset which has $(x_1, c_1), (x_2, c_2), ..., (x_n, c_n)$ without consideration of c_i. After this in class assignment stage, ARSkNN is used to find the k-nearest neighbors (instances) in D with respect to a query instance.

For a query instance y, $Massim^h(x, y)$ is estimated for all instances x in dataset D. For this estimation, x and y are parsed through t similarity trees (sTrees) of the similarity forest (sForest) After this in class assignment stage, ARSkNN is used to find the k-nearest neighbors (instances) in D with respect to a query instance.

Algorithm 1. ARSkNN

```
Input: $y$ — Query instance, $D$ — Dataset which has
        $\{(x_1, c_1), (x_2, c_2), ..., (x_n, c_n)\}$, $k$ — number of nearest neighbors
Output: $c_y$ — Class of query instance y
Let $A \leftarrow \{\}$;
for each $x$ in $D$ do
    $Massim \leftarrow Massim(x_i, y, F, e)$ ;
    $A \leftarrow A \cup \{x_i, c_i, Massim\}$;
end
Sort in ascending order, the pairs in $A$ using the third components;
$c_y \leftarrow$ the most frequent class in [Select the first $k$ instances from $A$];
return $c_y$
```

5 Performance Evaluation

The works of [24,25] reveals that error rates were often used as the measurement of classification accuracy of models. However, most records in the data set of credit card customers are non-risky (87.88%); therefore, the error rate is insensitive to classification accuracy of models.

It also varies from application to application in which classification technique is used. For the binary classification problem, some researchers have been used accuracy percentage for comparing the performance of different models than the error rate [26,27].

We have also demonstrated the average runtime of both classification techniques in seconds to justify that ARSkNN is very much faster than the traditional k-NN classifier, which uses a distance based similarity measure.

As every time with Java code, to get judiciously precise measurements, we are needed to exercise the significant code a few times before considering any measurement, so that the JIT has essentially compiled the code. To fulfil this purpose, we run each experiment 10 times and all the experiments was done with 10-fold cross-validation technique to evaluate both the classification techniques. In k-fold cross-validation, the original sample is arbitrarily subdivided into k equal size subsamples. Then of the k subsamples, a particular subsample is reserved as the validation data for analysis the model, and the left over k-1 subsamples are used as training data. The cross-validation method is then reiterated k times (folds), with each one of the k subsamples used precisely one time as the validation data. The k outcomes from the folds can then be averaged to calculate a single estimation. The benefit of this method is that all are used for both validation and training, and each interpretations is used for validation precisely one time.

5.1 Accuracy Percentage

In binary classification, accuracy is statistical measure which tells about how well a binary classifier correctly classifies the instances. According to ISO 5725-1, (Reference BS ISO 5725-1) the overall term "accuracy" is used to define the nearness of a quantity to the true value. It is the number of correct predictions made divided by the total number of predictions made, multiplied by 100 to turn it into a percentage i.e.

$$AccuracyPercentage = ((TP + TN)/All) * 100 \tag{1}$$

where, TP = True Positive; TN = True Negative and All = total number of instances.

5.2 Average Runtime

The analysis of algorithms is the assurance of the quantity of resources (such as storage and time) required to execute them. To evaluate the time complexity of any algorithm, calculation of runtime of that algorithm has to be done. In weka, the average runtime of classification technique can be calculated using the Experimenter module. One can use UserCPU_Time_training (in seconds) and UserCPU_Time_testing (in seconds) fields to output the average time for the classifiers in the experiment.

6 Description of the Dataset

The data had been collected from a bank in Taiwan which provides credit cards to its customers. The default of credit card clients data set is made of 30000 instances and 24 attributes along with class attribute, which is a binary variable (Yes = 1, No = 0). Among these 30000 instances, 6636 instances are the card owners with default payments.

In the very first study done on this dataset has shown comparative study of six data mining techniques (k-nearest neighbor, logistic regression, discriminant analysis, naive bayes classifier, artificial neural networks, and classification trees).

The description of 23 explanatory attributes are as follows:

ATT1: Credit Amount (in NT dollars).

ATT2: Sex (1 = male; 2 = female).

ATT3: Education (1 = grad. school; 2 = university; 3 = high school; 4 = others).

ATT4: Marital status (1 = married; 2 = single; 3 = others).

ATT5: Age (in years).

ATT6 - ATT11: History of past 6 payment. The measuring scale for the repayment status is: -1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; . . .; 8 = payment delay for eight months; 9 = payment delay for nine months and above.

ATT12 - ATT17: Amount of bill statement (in NT dollars).

ATT18 - ATT23: Amount of previous payment (in NT dollars).

7 Evaluation

This section presents the experimental results obtained by evaluating the performance of the ARSkNN classification technique against the traditional kNN classification technique. The whole classification experiment has been carried out by using a machine with an Intel Core i7 processor with 2.4 GHz speed and 8 GB RAM. The experiments were done using experimenter module of Weka 3.7.10 for both the classifiers. Traditional kNN classifier is already implemented in Weka with the name of IBK. In this experiment, IBK is used with LinearNNSearch as a nearest neighbor search algorithm with Euclidean distance as similarity measure. ARSkNN is implemented using Java Development Kit 1.8.0 with Netbeans 8.0.2 as the preferred IDE. The jar file titled ARSkNN.jar has been combined as a runtime module with Weka 3.7.10 platform.

Table 1 shows the obtained results in term of average accuracy percentage for IBK and ARSkNN (with 10, 50, and 100 sTrees) over 10-fold cross validation for different values of k which are 1, 3, 5 and 10. For the value of k as 1, ARSkNN has 8.22% gain in average accuracy in comparision to IBK, which is a huge gain in the classification domain. The same variation in results have been seen with the values of k as 3, 5 and 10.

The overall conclusion that can be drawn from Table 1 is, ARSkNN gives better average classification accuracy for every value of k. Figure 1 shows the same in the graphical form.

Table 1. Average accuracy (in percentage)

	k = 1	k = 3	k = 5	k = 10
IBK	72.97	77.69	79.33	80.76
ARSkNN with 10 sTrees	78.70	80.69	80.88	80.90
ARSkNN with 50 sTrees	80.98	81.38	81.33	81.14
ARSkNN with 100 sTrees	**81.19**	**81.44**	**81.39**	**81.25**

Table 2 shows the obtained results in term of average runtime in seconds for IBK and ARSkNN with 10, 50, and 100 sTrees over 10-fold cross validation for different values of k which are 1, 3, 5 and 10. For the value of k as 1, even ARSkNN with 100 trees has taken 6.81 seconds lesser than IBK. As we increase the value of k, this difference becomes even more prominent.

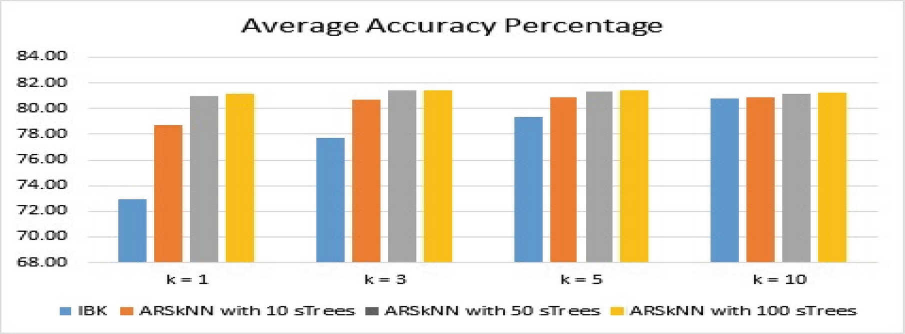


Fig. 1. Average Accuracy (in percentage) of classifiers.

Table 2. Average runtime (in seconds)

	k = 1	k = 3	k = 5	k = 10
IBK	7.80	9.71	10.41	10.30
ARSkNN with 10 sTrees	0.08	0.08	0.08	0.08
ARSkNN with 50 sTrees	0.46	0.52	0.56	0.72
ARSkNN with 100 sTrees	0.99	1.07	1.04	1.06

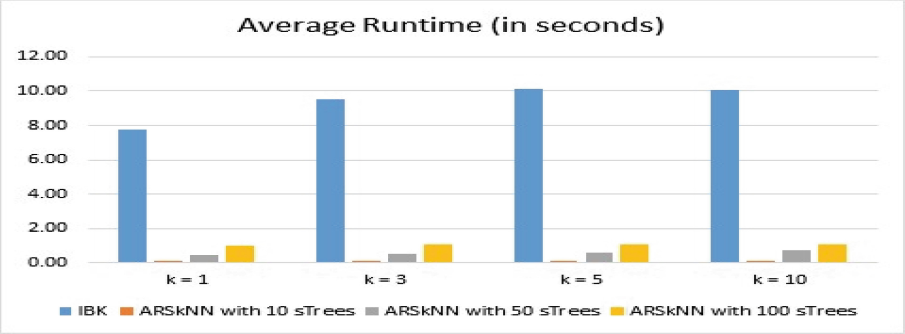


Fig. 2. Average Runtime (in seconds) of classifiers.

The overall conclusion that can be drawn from Table 2 is, ARSkNN gives significantly better average runtime for every value of k and it has also been shown in graphical form in Fig. 2.

8 Discussion

The research work assessed that ARSkNN is expressively improved than the IBK (with Euclidean distance) upon two parameters, i.e. accuracy percentage and average runtime. The accuracy percentage of IBK significantly depends upon the similarity measure used as learning metric.

ARSkNN is taking very less average runtime than the IBK because it computes the similarity measures during the modelling stage of sForest however IBK computes the similarity between each training instance and the testing instance which is an overhead for IBK. Also due to this computation overhead, IBK needs all the training instances in the memory, whereas ARSkNN does not needs any training instance in memory, which affects overall accuracy in the computed results at the end of the simulation.

After finding the k-nearest neighbors, both classifiers has to use the voting technique to decide the class of testing instance. The core difference between these classifiers can be seen in terms of the similarity measures via distance calculation in traditional kNN (IBK) and the mass based estimation in ARSkNN.

9 Conclusion

In this paper, we established ARSkNN classifier which uses similarity measure based upon mass estimation technique and demonstrate its effectiveness for the credit card risk analysis. The method was compared with traditional kNN technique on the credit data set, which shown the significant results in terms of the average accuracy percentage and average runtime. The modelling method was the major concerns as communicated in the paper. For the ARSkNN, the similarity Forest developed during the training phase is adopted to identify the similarity on the testing data set, which further acts as a class identifier during the voting phase in k-Nearest Neighbor measure. However, in case of traditional kNN the complete training set is used to calculate the distance metric during the testing phase for identifying k-Nearest Neighbor, which further voted for the class identification.

There are potential extensions of the current work. First, one can compare ARSkNN with kNN using different similarity metrics rather than Euclidean distance. Second, the application of ARSkNN should also be judged in various different domains.

References

1. Safakli, O.V.: Credit risk assessment for the banking sector of northern cyprus. Banks Syst. **2**(1), 21 (2007)
2. Bekiroglu, B., Takci, H., Ekinci, U.C.: Bank credit risk analysis with Bayesian network decision tool. IJAEST Int. J. Adv. Eng. Sci. Technol. **1**(9), 273–279
3. Karaa, A., Krichene, A.: Credit-risk assessment using support vectors machine and multilayer neural network models: a comparative study case of a tunisian bank. Account. Manag. Inf. Syst. **11**(4), 587 (2012)
4. Fisher, R.A.: The use of multiple measurements in taxonomic problems. Ann. Eugenics **7**(2), 179–188 (1936)
5. Durand, D., et al.: Risk Elements in Consumer Instalment Financing. NBER Books (1941)

6. Altman, E.I.: Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. J. Finance **23**(4), 589–609 (1968)
7. Steenackers, A., Goovaerts, M.J.: A credit scoring model for personal loans. Insur. Math. Econ. **8**(1), 31–34 (1989)
8. Hand, D.J., Henley, W.E.: Statistical classification methods in consumer credit scoring: a review. J. Roy. Stat. Soc. Ser. A (Stat. Soc.) **160**(3), 523–541 (1997)
9. Desai, V.S., Crook, J.N., Overstreet, G.A.: A comparison of neural networks and linear scoring models in the credit union environment. Eur. J. Oper. Res. **95**(1), 24–37 (1996)
10. Matoussi, H., Abdelmoula, A., et al.: Using a neural network-based methodology for credit-risk evaluation of a Tunisian bank. Middle East. Finance Econ. **4**, 117–140 (2009)
11. Davis, R.H., Edelman, D., Gammerman, A.: Machine-learning algorithms for credit-card applications. IMA J. Manag. Math. **4**(1), 43–51 (1992)
12. Miguéis, V.L., Benoit, D.F., Van den Poel, D.: Enhanced decision support in credit scoring using Bayesian binary quantile regression. J. Oper. Res. Soc. **64**(9), 1374–1383 (2013)
13. Foust, D., Pressman, A.: Credit Scores: Not-so-Magic Numbers. Business Week 7 (2008)
14. Went, P., Apostolik, R., Donohue, C.: Foundations of banking risk: an overview of banking, banking risks, and risk-based banking regulation. Wiley, Hoboken (2009)
15. Van Gestel, T., Baesens, B.: Credit Risk Management: basic concepts: financial risk components, rating analysis, models, economic and regulatory capital. Oxford University Press, UK (2009)
16. Guo, Y., WU, C.: Research on credit risk assessment in commercial bank based on information integration. In: Proceedings of 2009 International Conference on Management Science and Engineering (2009)
17. Zhang, D., Huang, H., Chen, Q., Jiang, Y.: A comparison study of credit scoring models. In: Third International Conference on Natural Computation, ICNC 2007, vol. 1, pp. 15–18. IEEE (2007)
18. Berry, M.J., Linoff, G.: Data Mining Techniques: For Marketing, Sales, and Customer Support. Wiley, Hoboken (1997)
19. Cover, T., Hart, P.: Nearest neighbor pattern classification. IEEE Trans. Inf. Theory **13**(1), 21–27 (1967)
20. Weinberger, K.Q., Saul, L.K.: Distance metric learning for large margin nearest neighbor classification. J. Mach. Learn. Res. **10**, 207–244 (2009)
21. Wang, J., Neskovic, P., Cooper, L.N.: Improving nearest neighbor rule with a simple adaptive distance measure. Pattern Recogn. Lett. **28**(2), 207–213 (2007)
22. Noh, Y.K., Zhang, B.T., Lee, D.D.: Generative local metric learning for nearest neighbor classification. IEEE Trans. Pattern Anal. Mach. Intell. **401**, 106–118 (2017)
23. Kumar, A., Bhatnagar, R., Srivastava, S.: ARSkNN-A k-NN classifier using mass based similarity measure. Procedia Comput. Sci. **46**, 457–462 (2015)
24. Jain, A.K., Duin, R.P.W., Mao, J.: Statistical pattern recognition: a review. IEEE Trans. Pattern Anal. Mach. Intell. **22**(1), 4–37 (2000)
25. Nelson, B.J., Runger, G.C., Si, J.: An error rate comparison of classification methods with continuous explanatory variables. IIE Trans. **35**(6), 557–566 (2003)
26. Jiawei, H., Kamber, M.: Data mining: concepts and techniques. Morgan Kaufmann, San Francisco (2001)
27. Witten, I.H., Frank, E., Hall, M.A., Pal, C.J.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, Burlington (2016)

Dental Age Estimation in East Asian Population with Least Squares Regression

Jiang Tao[1](✉), Mufan Chen[2], Jian Wang[1], Lin Liu[3], Aboul Ella Hassanien[4], and Kai Xiao[2]

[1] Department of General Dentistry, Ninth People's Hospital,
Shanghai Jiao Tong University School of Medicine, Shanghai, China
taojiang_doctor@hotmail.com, 27jane@sjtu.edu.cn

[2] Shanghai Jiao Tong University, Shanghai, China
{mfchen,showkey}@sjtu.edu.cn

[3] School of Public Health, Shanghai Jiao Tong University School of Medicine, Shanghai, China
handsomedog@sjtu.edu.cn

[4] Cairo University, Giza, Egypt
abo@egyptscience.net

Abstract. The purpose of the study is to derive a machine learning method of estimating overall dental maturity or dental age to achieve higher accuracy than the former methods. We select 1697 orthopantomograms of 877 boys and 820 girls from Shanghai Ninth People's Hospital, Shanghai Jiao Tong University School of Medicine, of which 1695 (875 boys and 820 girls) are used for estimating the dental age with the Demirjian's method, Willem's method and our method. Analysis of Variance (ANOVA) of randomized block design is performed to search for statistically significant differences between the chronological age and the dental age of three methods. Root-mean-square error (RMSE) is also performed in order to analyze the errors of the three methods. The adapted method is validated and results in more accurate dental age estimations in this population compared with Demirjian's method and Willem's method.

Keywords: Dental age estimation · Least squares estimation
Machine learning · Artificial intelligence · Regression

1 Introduction

Age estimation has been a prevailing method of age information retrieval in clinic dentistry [1,2] and forensic science [3,4]. A person's age information is vital under a variety of circumstances such as deciding criminal responsibility, illegal immigrants, social benefits and identifying deceased person's age while one's actual chronological age certificate is unknown or not provided [1,3,5–9]. Dental age, compared with other age estimation methods such as bone age estimation, is considered to be more accurate because tooth development shows

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 653–660, 2018.
https://doi.org/10.1007/978-3-319-74690-6_64

less variability than other developmental features and low variability in relation to chronological age [10,11]. Hence, dental age is considered vital in ascertaining the age of an individual.

Methods for estimating human dental age can be roughly divided into three categories: Morphological observation, biochemical analysis and radiological methods.

Morphological methods are based on assessment of teeth. Hence, these methods require extracted teeth for microscopic preparation or observing teeth in the mouth. Meanwhile, tooth eruption is affected by various local factors, such as crowding, extractions, ankylosis, ectopic positions, and persistence of primary teeth [10].

Biochemical analysis is based on the racemization of amino acids, which is a reversible first-order reaction and is relatively rapid in living tissues in which metabolism is slow. Aspartic acid has been reported to have the highest racemization rate of all amino acids and to be stored during aging. In particular, L-aspartic acids are converted to D-aspartic acids, thus the levels of D-aspartic acid in human enamel, dentine, and cementum increase with age [12].

Radiological methods play an indispensable role in the human age determination. Radiological images are utilized in the process of age estimation, which is one of the essential tools in identification in forensic science. Radiographic assessment of age is a simple, non-invasive and reproducible method that can be employed both on living and unknown dead. Radiological age estimation by tooth formation is considered as a more reliable indicator of dental maturity than that by tooth eruption [13].

Demirjian's method, as one of the most popular radiological methods, which was put forward in 1973, estimated chronological age based on developments of seven teeth from the left side of the mandible. This method was similar to Tanner-Whitehouse's method [14], which estimated chronological age based on the maturity of hands and wrists.

Demirjian, Goldstein, and Tanner used the stages that have usually been marked by recognizable tooth shapes, from the beginning of calcification through to final mature form. Useful stages must be easily recognizable. Since the stages are indicators of maturity but not the size, they cannot be defined by any absolute length measurements. So a system was built on eight calcification stages from calcification of the crown cusps to closure of the apex of the seven left permanent mandibular teeth. The stages were written in letters A-H representing the ordinal of tooth development. The score of each stage was assigned and the aggregate of the scores turned into the subject's dental maturity score (DMS). Acquired dental maturity score was then converted into dental age according to the provided tables.

The first advantage that makes the method proposed by Demirjian et al. [8] widely applied is its simpleness, i.e. it is an orthopantomogram based method that enables more reliable standardization and has good reproducibility and intra-examiner/inter-examiner reliability. The maturity scoring system within the method is universal in application, thus further enhances its acceptance scope, although the conversion to dental age depends on the population being considered. Furthermore, this conversion can be made with the use of relatively small local samples and can reach a relatively accurate result.

However, there are some limitations. First, Demirjian's method uses orthopantomograms which are difficult to obtain in young children, due to technical reasons, as well as legal and ethical considerations. Second, since simultaneous evaluations of seven left mandibular teeth are required, it cannot be applied in children with lacking teeth inborn or acquired. It is reported that Demirjian's method resulted in a consistent overestimation of the dental age for the first Belgian Caucasian sample, amounting to a median of 0.5 years for boys and a median of 0.6 years for girls [15]. Third, this method may not express agenesis of teeth, distinctive retardation of dental development (excluding third molars), and systemic diseases and various developmental stages of the tooth. Forth, the appreciation of developmental stage may become difficult as the choice of the tooth developmental stage is quite subjective. Fifth, this method does not give maturity scores for stages 1–4 in case of 1st molar, central and lateral incisor, thus excluding the individuals below the age of 4–4.5 years.

Although the modified conversion table by Willem et al. [15] resolves some of the aforementioned issues in the Demirjian method, it can be seen that the technique is proposed based on clinical experience with limited validation, hence the results are not consistent within different databases for Demirjian's and Willem's method [16–23].

It can be clearly seen that, two issues lied in the existing techniques of Demirjian methods. First, the sample population will be limited to one specific population and the change of sample incurs modification of the model. Second, the model is a fixed one without possibility of automatic tuning. In this paper, we propose one method with the use of machine learning techniques based on the studied factors adapted from Demirjian's methods, including the pioneering articles in this field, the breakthroughs, the current situation, the most influential articles so far.

The aim of the present study is to derive a machine learning method of estimating overall dental maturity or dental age which has never been used before to achieve higher accuracy than the former methods that are proposed based on experience.

2 Materials and Methods

Regression measures relationships between input variables and continuous output variables from existing data sets, and makes predictions for new inputs. Each regression algorithm assumes a certain type of model. Here are some of the benefits of using regression analysis: it indicates relationship between the independent variables and the dependent variables; it indicates the effect of multiple arguments on the dependent variable. Regression analysis allows us to compare variables of different scales, which helps data analysts to remove and evaluate variables used to build predictive models. That is to say, we use curves to fit these data points to minimize the distances between points and curves.

Regression models can generally be divided into two groups: linear regression and non-linear regression. The classic algorithm for linear regression is linear least squares. In that case, assume $X = [x_1, x_2, ..., x_M]^\mathsf{T}$, with each component representing the input of a feature. The relationship between input and output can be represented as

$$d = \sum_{i=1}^{M} w_i x_i + \epsilon$$

where d is a corresponding input variable and M is the number of input variables. $w_1, w_2, ..., w_i$ define a set of parameters. The above expression can also be written in the form of matrix:

$$d = \mathbf{W}^\mathsf{T} + \epsilon.$$

The cost function or the sum of squared residuals can be defined as

$$J = \frac{1}{2}\sum_{i=1}^{M}(d_i - d)^2$$

where the training example data set is denoted as $\{(x_1, d_1), (x_2, d_2), ..., (x_M, d_M)\}$. Linear least squares (LLS) uses the sum of squares of the residuals as the error measurement to minimize. In the model of Regularized Linear Least Squares, The cost function becomes

$$J = \frac{1}{2}\sum_{i=1}^{M}(d^i - d)^2 + \frac{\lambda}{2}X^2$$

where λ is a parameter that determines the trade-off between small values in a and small residuals [24].

However, in many practical problems, the regression function is often a more complex nonlinear function. Linear least squares regression can be extended to non-linear regression in two distinct ways, yielding two classes of algorithms [24]: Algorithms that perform multiple weighted LLS regression, using different input-dependent weighting functions, such as Locally Weighted Regression (LWR) [25]; Algorithms that project the input space into a feature space using a set of non-linear basis functions, and performing one LLS regression in this projected feature space, such as Support Vector Regression (SVR) [26]. LLS regression method has been selected in this paper due to its simplicity and stability.

The sample of the population consists of 1697 orthopantomograms or panoramic radiographs of 820 boys (age ranging from 10.73 to 18.98 years) and 877 girls (age ranging from 11.01 to 19.00 years), which are selected from patients' records of Shanghai Ninth People's Hospital, Shanghai Jiao Tong University School of Medicine. The mean and standard derivation (SD) of the chronological age of boys and girls are 14.93 ± 2.30 and 15.03 ± 2.32 respectively. Exclusion criteria are: age above 19.00 years at the time the orthopantomogram is taken; no seven permanent teeth or the chronological age; systemic diseases; premature birth; congenital anomalies; unclear orthopantomogram; aplasia of at least two corresponding teeth bilaterally in the mandible. To achieve intra-examiner reliability, we invite six people with careful training to watch the Dental Panoramic Tomograph (DPT) and evaluate the stages separately. We then take the mode of six people's evaluation as the final result. Of this sample, 1695 orthopantomograms (820 boys and 875 girls) could be used for estimating the dental age.

Dental age estimations are performed with the Demirjian's method, Willem's method and our method. Both Demirjian and Willem use a notion of index in their conversion table, which is similar to the notion of feature in machine learning method. So we consider the Tooth Development Stages (TDS) of seven permanent teeth as seven features and use data of certain population to find the relationship behind it. Stage A to Stage H corresponds to number one to eight respectively. We use least squares regression [27] to fit the data. Based on

$$\sum_{i=1}^{7} a_i * w_i + b = t$$

where a_i is the rating data of seven teeth respectively and w_i and b are weights and biases we want to fit in, we use 5-fold cross validation to fit our data.

To compare the accuracy of three methods, each subject is regarded as a block, the different calculation methods are regarded as treatments, the total difference among the three methods is analyzed using the Analysis of Variance (ANOVA) of randomized block design, then the differences between every two treatment groups are analyzed using the Student-Newman-Keuls (SNK) method (SAS Statistical Software Package, SAS Institute, Cary, NC), P value less than 0.05 indicates that the difference is statistically significant [28].

We also use root-mean-square error (RMSE) [29], which is

$$E = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(t-s)^2}$$

where s is the actual age, t is the estimated value and n is the number of data entries.

3 Experiments and Results

The data (shown in Table 1) consists of panoramic radiographs of the teeth of 820 boys and 875 girls aged 10 to 20, examined in Shanghai Ninth People's Hospital, Shanghai Jiao Tong University School of Medicine (Jiang Tao et al. 2017). We took radiographs only from children free from any disorder affecting growth and who had a complete mandibular permanent dentition (erupted or not). All had parents and grand-parents of East Asian origin.

Girls and boys were treated separately because of sex-tooth interaction which is a tooth is relatively more advanced in one sex than the other. This is known that it occurs when teeth erupt, and so do our scores, since girls' are higher than boys' in all teeth except M1 where girls' are lower.

Table 1. Number of children at every two age groups

Age	Male	Female	Total
10˜	84	113	197
12˜	239	204	443
14˜	210	222	432
16˜	180	225	405
182̃0	107	111	218
Total	820	875	1695

The adapted scoring system for dental age estimation resulted in more accurate predictions than Demirjian's method and Willem's method. There are statistically significant differences of D-values between the chronological age and the dental age of three methods for boys ($F = 36.65$, $p < 0.0001$) and girls ($F = 50.32$, $p < 0.0001$) analyzed by the ANOVA of randomized block design. The Demirjian's method resulted in an error-estimation for boys (mean: 1.052, SD: 1.701) and girls (mean: 1.344, SD: 1.971). The Willem's method resulted in an error-estimation for boys (mean: 1.585, SD: 1.670) and girls (mean: 1.633, SD: 1.866). Our method resulted in an error-estimation for boys (mean:0.029, SD: 1.795) and girls (mean: 0.026, SD: 2.009) as shown in Table 2 and Fig. 1.

Table 2. The differences between the chronological age and the dental age of three methods for boys and girls (mean±s.d.)

Method	Male	Female
D's method [8]	1.052 ± 1.701	1.344 ± 1.971
W's method [15]	1.5854 ± 1.670	1.633 ± 1.866
Our method	0.029 ± 1.795	0.026 ± 2.009

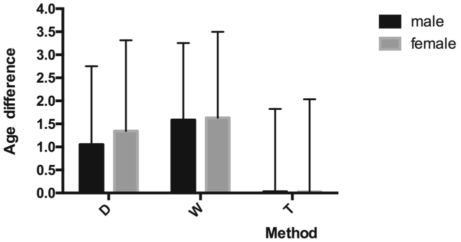


Fig. 1. The age differences among three methods for male and female

The errors caused by the three methods are analyzed by RMSE.The errors caused by the Demirjian's method was 1.998 for boys and 2.384 for girls.

The errors caused by the Willem's method was 2.302 for boys and 2.485 for girls. The errors caused by our method are 1.794 for boys and 2.008 for girls which are less than the two methods mentioned above as shown in Table 3.

Table 3. Errors of our method compared with D's and W's

Method	Male	Female
D's method [8]	1.998	2.384
W's method [15]	2.302	2.485
Our method	1.794	2.008

4 Discussion

Error-overestimations of chronological age when using the method reported by Demirjian and Willem are mostly found. We initially use Demirjian's method and Willem's method in the scoring system. They are well known but there are some limitations mentioned above, we therefore investigate whether our method would give more accurate estimation.

1697 patient's records are selected from Shanghai Ninth People's Hospital, Shanghai Jiao Tong University School of Medicine. Of this sample 1695 orthopantomograms (875 boys and 820 girls) could be used for estimating the dental age. We calculate the scores of the seven teeth from the left side with Demirjian's method, Willem's method and our method. For each child we estimate the dental age with three methods and then take the differences between the chronological age and the dental age of three methods respectively. There are statistically significant differences for boys and girls separately analyzed by the ANOVA of randomized block design. The errors caused by the three methods are analyzed by RMSE. It appears that our method is acceptable and more accurate.

5 Conclusion

This study shows statistically significant less error-estimation of the dental age using our method. The adapted method shows more accurate dental age estimations in this population compared with Demirjian's method and Willem's method, but may not be valid in other populations. In the future, we will go on exploring other machine learning techniques to achieve higher accuracy and investigating accuracy on other populations. Strength and weakness of LLS regression method will be reported along with comparative results of other regression methods.

References

1. Panchbhai, A.: Dentomaxillofac. Radiol. **40**(4), 199 (2011)
2. Liversidge, H.: Int. J. Paediatr. Dent. **9**(2), 111 (1999)
3. Ritz-Timme, S., Cattaneo, C., Collins, M., Waite, E., Schütz, H., Kaatsch, H.J., Borrman, H.: Int. J. Legal Med. **113**(3), 129 (2000)
4. Schmeling, A., Reisinger, W., Geserick, G., Olze, A.: Forensic Sci. Med. Pathol. **1**(4), 239 (2005)
5. Liversidge, H.M.: Arch. Oral Biol. **45**(9), 713 (2000)
6. Hunt, E.E., Gleiser, I.: Am. J. Phys. Anthropol. **13**(3), 479 (1955)
7. Nolla, C.M., et al.: The development of permanent teeth. Ph.D. thesis, University of Michigan (1952)
8. Demirjian, A., Goldstein, H., Tanner, J.: Hum. Biol. **45**, 211–227 (1973)
9. Patnana, A.K., Vabbalareddy, R.S., Vanga, N.R.V.: Int. J. Clin. Pediatr. Dent. **7**(3), 186 (2014)
10. Leurs, I., Wattel, E., Aartman, I., Etty, E., Prahl-Andersen, B.: Eur. J. Orthod. **27**(3), 309 (2005)
11. Pelsmaekers, B., Loos, R., Carels, C., Derom, C., Vlietinck, R.: J. Dent. Res. **76**(7), 1337 (1997)
12. Elfawal, M.A., Alqattan, S.I., Ghallab, N.A.: Med. Sci. Law **55**(1), 22 (2015)
13. Bagherian, A., Sadeghi, M.: J. Oral Sci. **53**(1), 37 (2011)
14. Bull, R., Edwards, P., Kemp, P., Fry, S., Hughes, I.: Arch. Dis. Child. **81**(2), 172 (1999)
15. Willems, G., Van Olmen, A., Spiessens, B., Carels, C.: J. Forensic Sci. **46**(4), 893 (2001)
16. Wolf, T.G., Briseño-Marroquín, B., Callaway, A., Patyna, M., Müller, V.T., Willershausen, I., Ehlers, V., Willershausen, B.: BMC Oral Health **16**(1), 120 (2016)
17. Gupta, S., Mehendiratta, M., Rehani, S., Kumra, M., Nagpal, R., Gupta, R.: J. Forensic Dent. Sci. **7**(3), 253 (2015)
18. Flood, S.J., Franklin, D., Turlach, B.A., McGeachie, J.: J. Forensic Legal Med. **20**(7), 875 (2013)
19. Zhai, Y., Park, H., Han, J., Wang, H., Ji, F., Tao, J.: J. Forensic Legal Med. **38**, 43 (2016)
20. Urzel, V., Bruzek, J.: J. Forensic Sci. **58**(5), 1341 (2013)
21. Balla, S.B., Baghirath, P.V., Vinay, B.H., Kumar, J.V., Babu, D.G.: J. Forensic Legal Med. **43**, 21 (2016)
22. Duangto, P., Janhom, A., Prasitwattanaseree, S., Mahakkanukrauh, P., Iamaroon, A.: Forensic Sci. Int. **266**, 583 (2016)
23. Ye, X., Jiang, F., Sheng, X., Huang, H., Shen, X.: Forensic Sci. Int. **244**, 36 (2014)
24. Stulp, F., Sigaud, O.: Neural Netw. **69**, 60 (2015)
25. Atkeson, C.G., Schaal, S.: Neurocomputing **9**(3), 243 (1995)
26. Vapnik, V.N., Vapnik, V.: Statistical Learning Theory, vol. 1. Wiley, New York (1998)
27. Leon, S.J.: Linear Algebra with Applications. Macmillan, New York (1980)
28. Littell, R.C.: SAS. Wiley Online Library (1996)
29. Hyndman, R.J., Koehler, A.B.: Int. J. Forecast. **22**(4), 679 (2006)

Text Mining Approach to Analyse Stock Market Movement

Mazen Nabil Elagamy[1,2(✉)], Clare Stanier[1], and Bernadette Sharp[1]

[1] Staffordshire University, Staffordshire, Stoke-on-Trent ST4 2DE, UK
[2] Arab Academy for Science and Technology, Abo Keer, Alexandria 1029, Egypt
m_elagami@hotmail.com

Abstract. Stock Market (SM) is a significant sector of countries' economy and represents a crucial role in the growth of their commerce and industry. Hence, discovering efficient ways to analyse and visualise stock market data is considered a significant issue in modern finance. The use of Data Mining (DM) techniques to predict stock market has been extensively studied using historical market prices but such approaches are constrained to make assessments within the scope of existing information, and thus they are not able to model any random behaviour of stock market or provide causes behind events. One area of limited success in stock market prediction comes from textual data, which is a rich source of information and analysing it may provide better understanding of random behaviours of the market. Text Mining (TM) combined with Random Forest (RF) algorithm offers a novel approach to study critical indicators, which contribute to the prediction of stock market abnormal movements. A Stock Market Random Forest-Text Mining system (SMRF-TM) is developed to mine the critical indicators related to the 2009 Dubai stock market debt standstill. Random forest is applied to classify the extracted features into a set of semantic classes, thus extending current approaches from three to eight classes: critical down, down, neutral, up, critical up, economic, social and political. The study demonstrates that Random Forest has outperformed the other classifiers and has achieved the best accuracy in classifying the bigram features extracted from the corpus.

Keywords: Knowledge discovery · Text mining · Natural language processing Stock market · Random forest

1 Introduction

Knowledge discovery is a fast-growing field of research providing hidden and valuable knowledge stored in ever increasing amounts of data. We have rich and readily available sources of data and texts, whether stored in databases, newspapers, or in other scientific and business repositories. This has created the urgent need for novel computational theories and tools to analyse and extract valuable hidden insights from this explosive growth of digital data. Data mining, which extracts knowledge from structured datasets, and text mining which analyses unstructured documents, are subfields of knowledge discovery.

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 661–670, 2018.
https://doi.org/10.1007/978-3-319-74690-6_65

Text mining is the focus of this paper aimed at demonstrating its potential and valuable contribution to stock market crashes analysis, which is an important event of today's global economy. Countries around the world depend on stock markets for their economic growth. Stock market crashes are unavoidable and are, by nature, preceded by speculative economic bubbles. The increasing importance of stock markets and their direct influence on the economy were the main reasons for deciding to study and analyse stock market crashes, which is the application domain of our research. The main focus of this research project is to text mine the 2009 Dubai stock market debt standstill. Some crashes, such as the 1929 Wall Street crash, can often be difficult to collect sufficient textual data (financial news) suitable for deep analysis. Stock market movements can be specific to particular economies and political environments such as the 1973–1974 United Kingdom stock market crash, the 1998 Russian financial crisis and the Chinese stock bubble of 2007. In 2009, a number of factors contributed to the United Arab Emirates crisis; these include the global recession, the bursting of the Dubai property bubble, and the post Lehman shutdown of international capital markets hit simultaneously. Dubai witnessed a significant slowdown in growth and strains in its banking system as a result of the global financial crisis, the decline in oil prices, and in particular the bursting of its property bubble [1].

This paper is organised as follows. Section 1 has introduced the background of our study. Section 2 reviews related work on the study of financial applications. Section 3 describes the approach developed to analyse documents related to the 2009 Dubai stock market debt standstill and to discover the critical indicators associated with this crisis. Section 4 discusses the findings from our investigation and Sect. 5 presents conclusions and suggestions for future work.

2 Related Work

Financial data analysis has traditionally dealt with large volumes of structured data reflecting economic performance. However, the behaviour of the market is dictated by contemporary local and global events, which are not captured in the statistical data [2]. Data mining research projects have made use mainly of quantifiable information in terms of charts or numeric time series, which only describe the event but not their causes [3]. There is still a major problem with prediction approaches based only on historical market prices; this is the ability to model and explain any random behaviour of the market [4].

Textual data such as news, financial reports and economic articles are an important source of information describing stock market events; their analysis can provide a better understanding of any random behaviour of the market, which is difficult to be justified by focusing solely on historical and statistical data. Consequently, many researchers have explored the field of text mining to understand the causes of stock market movements in order to improve the prediction of its abnormal movements. Whilst tables with financial data indicate how well a company has achieved, the linguistic structure and written style of the text may tell more about its strategy and future performance [5].

The use of data mining techniques to analyse stock markets has been extensively studied using structured data like past prices, historical earnings, or dividends. On the other hand, text mining approaches are comparatively rare in this field due to the difficulty of extracting relevant information from unstructured data. [6] discussed the efficient stock market hypothesis, which states that stocks behave randomly. [4, 7] explain that the application of data mining to the analysis of stock market data using current approaches may not be sufficient to model and justify random behaviour of the market based only on quantitative data such as the values of stocks and historical market prices. This suggests that if researchers focus on the impact of unquantifiable events on the market, which can be extracted from related news articles, it may be possible to provide information about the random behaviour of the market and enhance analysis performance. [8] stated that there are huge amounts of free news and financial data, which are believed to contain rich information known as "alpha". The hypothesis of his research is that text mining approaches can be applied to enhance the performance of current trading systems' strategies if the "alpha" embedded in financial news is used to support the prediction of stock market share price movement directions [9].

Most textual sources used by text mining researchers for market prediction include financial journals and news such the Wall Street Journal, Financial Times, Reuters, Dow Jones, Bloomberg and even Yahoo Finance, and often the analysis is focused on the news text or the news headlines [10]. The literature review reveals two main text mining approaches are adopted in the analysis, prediction or mining of stock market features: (i) machine learning such as Support Vector Machine, decision rules/trees, regression algorithms, naïve Bayes, and (ii) natural language processing algorithms [11]. Machine learning algorithms give computers the ability to learn without being explicitly programmed by involving a set of data to train the algorithm and using another set of data to test the generated predictions. The natural language processing approach involves lexical, syntactic, semantic and pragmatic analysis of unstructured texts [11].

The study carried out by [4] reveals that automatic text classification techniques are commonly used in analysing incoming news, and in some cases researchers make use of historical market prices data related to stock price to improve the accuracy of their prediction, thus combining data and text mining algorithms. Such predictive systems consist of three main components: news labelling, classifier input generation, classification and finally news labelling. [12] propose a unified latent space model to characterise the "co-movements" between stock prices and news articles and to predict the closing stock prices on the same day; their algorithm is based on the analysis of daily articles from The Wall Street journal. [13] predict the stock market based on textual information from user-generated micro-blogs using the latent space model to correlate the movements of both stock prices and social media content. [14] apply natural language processing to analyse economic news articles of a media company to categorise and extract the sentiments and opinions expressed by the writers. Their aim is to identify the correlation between news and stock market fluctuations. A linguistic based text mining approach is adopted by [15] demonstrating how text mining could be integrated with the financial fraud ontology to improve the efficiency and effectiveness of extracting financial concepts. [16] analysed financial news articles and stock quotes covering the S&P 500 stock market index during a five week period using a set of linguistic textual representations, including bag of words, noun phrases, and named entities approaches to estimate a discrete stock price twenty minutes after a news article was released. [17] proposed a sentiment analysis system based on summarisation to determine the polarity (positive or negative) of news articles from the Wall Street Journal and financial market data from the NASDAQ aimed at predicting the stock market. In addition, [18] also constructed a predictive model to predict stock market future trends. Their model used sentiment analysis of multiple types of financial news and historical stock prices, which leaded to the achievement of prediction accuracy up to 89.80%.

3 Stock Market Random Forest–Text Mining (SMRF-TM)

This study has developed a text mining system, SMRF-TM, using a natural language processing approach to investigate the critical indicators that can contribute to the prediction of abnormal stock movements. The architecture of SMRF-TM consists of six main stages: textual data collection, documents pre-processing, features generation, features extraction, semantic analysis and classification of extracted features, and validation (see Fig. 1).

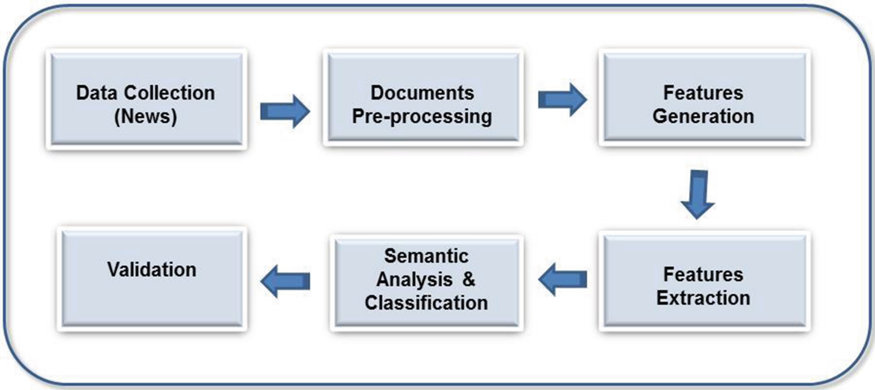


Fig. 1. SMRF-TM architecture

The data collection consisted of stock market articles related to Dubai Debt Standstill 2009. A total of 544 financial news articles concerning Dubai's stock market, published in the period between 2008 till 2012, were retrieved. This specific period was chosen so that it includes articles published before the crisis and after the crisis within the period for recovery of Dubai's SM (Dubai's SM upturn). These articles are used for training and testing.

Document pre-processing involves a number of lexical tasks but the most significant include data preparation, tokenisation, stop words removal and word stemming. Tokenisation generated 15,276 unigrams tokens and 103,506 bigrams features.

Features extraction reduces high-dimensionality by only selecting the most relevant features. Extracted textual features applied syntactic analysis to generate unigrams, bigrams, noun phrases, proper nouns or named entities. In SMRF-TM, the focus was on the generation and extraction of unigrams and bigrams. The Term Frequency/Inverse Document Frequency (TF/IDF) [19, 20] was applied and only words with the highest TF/IDF score were selected as significant features. A two-dimensional vector space was used to capture the relevant extracted features for each document within our data; the rows represented the document and the columns represented its features, and the cells captured the TF/IDF value for each feature.

Semantic and pragmatic analysis aims at discovering the relationships between these extracted features and classifying them into appropriate semantic classes: critical down, down, neutral, up and critical up, in contrast with current approaches, which use three main classes: down, neutral and up [4, 17, 21–23]. To discover the hidden knowledge and relations between the extracted features we applied the Random Forest (RF) algorithm, which runs efficiently on large data and is able to classify large amounts of data with accuracy. RF, which is an ensemble learning method, is a combination of tree predictors where each tree depends on the values of a random vector sampled independently with the same distribution for all trees in the forest [24]. The basic principle is that a group of "weak learners" can come together to form a "strong learner".

The RF application in SMRF-TM system generates an ensemble of 10 random, individual and un-pruned trees. Each individual tree is constructed using the following algorithm (Pseudo code):

```
Inputs: t (the number of random trees in the forest (iterations = 10))
        S (the training set)
        n (number of random features used in constructing each of the 10 trees)
Outputs: $T_t$ ; t =1,…., 10
        t = 1
        Do
        $s_t$ is a subsample articles from S with replacement
        Construct classifier tree $T_t$ using a decision tree inducer on $s_t$
        t ++
        while (t < 10)
```

The input parameter (n) represents the number of features, which is used to determine the decision at a node of the tree and it should be much less than the total number of features in the training set (S). The constructed ensemble decision trees (10 trees) are not pruned and the best split at each node is chosen from among the (n) random features not all the features. The classification of any unlabelled feature/news article is performed using the majority votes.

The first phase of our study focused on classifying the features into two classes, namely up and down; the aim was to evaluate the feasibility of our approach. Subsequently, the second phase extended the analysis to include five more classes: critical down, down, neutral, up and critical up; this has significantly enhanced the performance of SMRF-TM using unigram and bigram features. A better performance was achieved by implementing our approach on bigram features applied to a larger dataset, which is described in this paper.

In order to validate the results and achieve the best performance accuracy, a cross validation with different folds (5, 10, 20, 30, 40, and 50) were used to check which classifier yields the greatest classification performance. In cross validation, the training set is divided into mutually exclusive and equal sized subsets and for each subset the classifier is trained on the union of all the other subsets. The average of the error rate of each subset is therefore an estimate of the error rate of the classifier [25]. For example the 10-folds cross validation uses 9/10 of the data for training the algorithm and 1/10 of the data for testing, this process is repeated 10 times after shuffling the data, each time. In addition we have applied the Expectation Maximisation (EM) clustering algorithm which iteratively refines initial mixture model parameter estimates to better fit the data and terminates at a locally optimal solution [26]. In SMRF-TM, EM was applied to cluster the classified features and the news articles according to their semantic meanings in one of the three clusters: economic, social/geographical or political facet. Tables 1 and 2 provide an extract of the classification outputs for both unigram and bigram features, which are stemmed and their stems are not always the linguistic root of the words but related words map to the same stem. The application of EM clustering technique resulted in clustering 93% of the extracted features to the economical cluster, 5% to the social cluster and 2% to the political cluster. Such results are considered reasonable given the source of our dataset is retrieved from the Financial Times.

Table 1. Extract of clustered unigrams stemmed features

Economic facet	Social/Geographical facet	Political facet
Fund	Gulf	Govern
Invest	China	Vote
Return	Russia	Diplomat
Bond	Dubai	Elect
Sharehold	Middl	Regim
Financi	East	Parliament
Growth	Cultur	Polit
Stock	UK	Governor
Market	London	Conserve

Table 2. Extract of clustered bigrams stemmed features

Economic facet	Social/Geographical facet	Political facet
Price inflat	Niger delta	Press freedom
UK econom	Visit Vatican	Polici tighten
Global bond	People concern	Parliament result
Inflationary risk	People worri	New govern
Debt sustain	Rise unemploy	Democracy activist
Big risk	Britain strong	UAE democraci
Debt share	Nation insur	Conserve win
Risk economi	Help people	Weak govern
Help economi	Nation infrastructur	Conserve govern

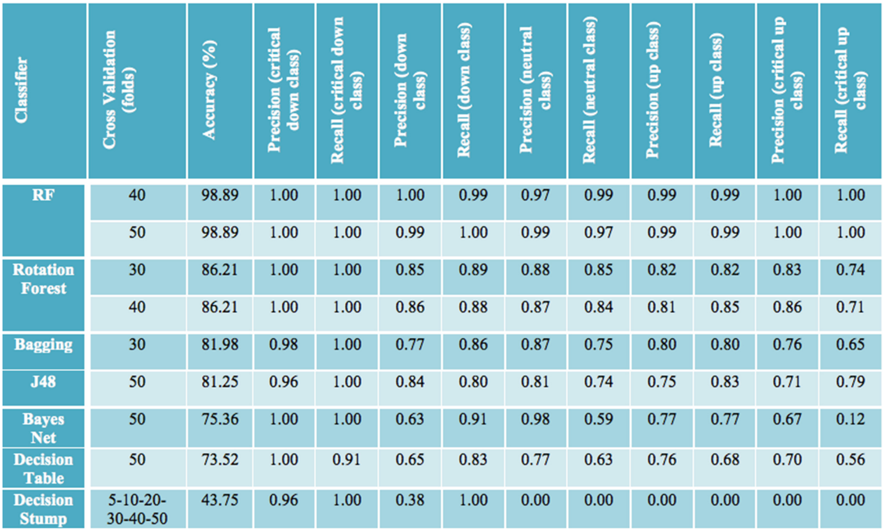
4 Discussion

In summary, the RF classifier has correctly classified 538 out of 544 documents resulting in 98.89% classification accuracy when applied using bigrams tokens. The neutral cluster has 132 articles true positive, four articles false positive and two articles false negative. The down class has 182 articles true positive, zero articles false positive and two articles false negative. The up class has 136 articles true positive, two articles false positive and two articles false negative. Regarding the critical down and the critical up classes they did not have any misclassified articles, hence, they have zero articles false positive and zero articles false negative.

To the best of our knowledge RF has not been applied to analyse stock market movements using textual data. Hence, we can only compare the classification performance of RF to the classification performance of other classifiers. The classification results produced by Random Forest were compared against the following ensemble of classifiers: Rotation Forest, Bagging, J48graft, Bayes Net, Decision Table, and Decision Stump. The main idea behind validation is to combine a set of models where each of these algorithms classifies the same original data, in order to achieve a better composite global model than using a single model to improve accuracy and thus obtain reliable estimates or decisions.

Table 3 summarises the best results obtained for each algorithm, expressed in terms of precision and recall measures. Random Forest achieved the best performance with 98.89% accuracy with 40 folds and decision stump achieved the worst performance overall with 43.75% accuracy measure (see Table 3). These measures are defined below.

Table 3. Summary of the best performance results of the 7 classifiers



Classifier	Cross Validation (folds)	Accuracy (%)	Precision (critical down class)	Recall (critical down class)	Precision (down class)	Recall (down class)	Precision (neutral class)	Recall (neutral class)	Precision (up class)	Recall (up class)	Precision (critical up class)	Recall (critical up class)
RF	40	98.89	1.00	1.00	1.00	0.99	0.97	0.99	0.99	0.99	1.00	1.00
	50	98.89	1.00	1.00	0.99	1.00	0.99	0.97	0.99	0.99	1.00	1.00
Rotation Forest	30	86.21	1.00	1.00	0.85	0.89	0.88	0.85	0.82	0.82	0.83	0.74
	40	86.21	1.00	1.00	0.86	0.88	0.87	0.84	0.81	0.85	0.86	0.71
Bagging	30	81.98	0.98	1.00	0.77	0.86	0.87	0.75	0.80	0.80	0.76	0.65
J48	50	81.25	0.96	1.00	0.84	0.80	0.81	0.74	0.75	0.83	0.71	0.79
Bayes Net	50	75.36	1.00	1.00	0.63	0.91	0.98	0.59	0.77	0.77	0.67	0.12
Decision Table	50	73.52	1.00	0.91	0.65	0.83	0.77	0.63	0.76	0.68	0.70	0.56
Decision Stump	5-10-20-30-40-50	43.75	0.96	1.00	0.38	1.00	0.00	0.00	0.00	0.00	0.00	0.00

$$Precision\,(PR) = \frac{True\,Positive\,(TP)}{True\,Positive\,(TP) + False\,Positive\,(FP)} \tag{1}$$

$$Recall\,(RC) = \frac{True\,Positive\,(TP)}{True\,Positive\,(TP) + False\,Negative\,(FN)} \tag{2}$$

$$Accuracy = \frac{Total\,number\,of\,correctly\,classified\,instances}{Total\,number\,of\,instances} * 100 \tag{3}$$

5 Conclusion

The need to determine early warning indicators for both banking and stock market crises has been the focus of study by many economists and politicians. Whilst most research projects into identifying these critical indicators applied data mining to uncover hidden knowledge, very few attempted to adopt a text mining approach. [6] explained that stock markets behave randomly; consequently the application of data mining to the analysis of stock market data may not be sufficient to model and justify any random behaviour of the market. Given the huge amounts of free news and financial data, it is important to study the rich information embedded in this data, known as “alpha”. This study is an attempt at addressing this issue and discovers the critical indicators from unstructured yet valuable source of information.

This paper has described a natural language processing driven approach at analysing documents related to the 2009 Dubai stock market debt standstill in order to mine these critical indicators. Random forest was applied to classify the extracted features into a set of semantic classes, thus extending current approaches from three to eight classes (critical down, down, neutral, up, critical up, economic, social and political). The study demonstrated that Random forest has outperformed the other classifiers and achieved the best accuracy in classifying the bigram features extracted from the selected corpus. It is proposed to apply this approach to other news articles and other stock market crises in order to refine the discovery of critical indicators.

References

1. IMF: United Arab Emirates 2009 Article IV Consultation—Staff Report; Public Information Notice; and Statement by the Executive Director for United Arab Emirates, IMF Country Report No. 10/42, February 2010
2. Gómez, M.M.Y., Gelbukh, A., López, A.L.: Mining the news: trends, associations, and deviations. Computación Sistemas **5**(1), 14–24 (2001)
3. Wüthrich, B., Permunetilleke, D., Leung, S., Lam, W., Cho, V., Zhang, J.: Daily prediction of major stock indices from textual WWW data. HKIE Trans. **5**(3), 151–156 (1998)
4. Nikfarjam, A., Emadzadeh, E., Muthaiyah, S.: Text mining approaches for stock market prediction. In: The 2nd International Conference on Computer and Automation Engineering ICCAE 2010, vol. 4, pp. 256–260. IEEE (2010)

5. Kloptchenko, A., Eklund, T., Karlsson, J., Back, B., Vanharanta, H., Visa, A.: Combining data and text mining techniques for analysing financial reports. Intell. Syst. Acc. Finan. Manag. **12**(1), 29–41 (2004)
6. Patel, J., Shah, S., Thakkar, P., Kotecha, K.: Predicting stock market index using fusion of machine learning techniques. Expert Syst. Appl. **42**(4), 2162–2172 (2015)
7. Schumaker, R.P., Zhang, Y., Huang, C.N., Chen, H.: Evaluating sentiment in financial news articles. Decis. Support Syst. **53**(3), 458–464 (2012)
8. Drury, B.: A text mining system for evaluating the stock market's response to news. Doctoral dissertation in Computer science, University of Porto (2013)
9. Kumar, B.S., Ravi, V.: A survey of the applications of text mining in financial domain. Knowl.-Based Syst. **114**, 128–147 (2016)
10. Nassirtoussi, A.K., Aghabozorgi, S., Wah, T.Y., Ngo, D.C.L.: Text mining for market prediction: a systematic review. Expert Syst. Appl. **41**(16), 7653–7670 (2014)
11. Gonçalves, P., Araújo, M., Benevenuto, F., Cha, M.: Comparing and combining sentiment analysis methods. In: 2013 Proceedings of the First ACM Conference on Online Social Networks, pp. 27–38. ACM (2013)
12. Ming, F., Wong, F., Liu, Z., Chiang, M.: Stock market prediction from WSJ: text mining via sparse matrix factorization. In: IEEE International Conference on Data Mining, ICDM 2014, pp. 430–439. IEEE (2014)
13. Sun, A., Lachanski, M., Fabozzi, F.J.: Trade the tweet: social media text mining and sparse matrix factorization for stock market prediction. Int. Rev. Financ. Anal. **48**, 272–281 (2016)
14. Kim, Y., Jeong, S.R., Ghani, I.: Text opinion mining to analyze news for stock market prediction. Int. J. Adv. Soft Comput. Appl. **6**(1) (2014)
15. Ali, M.M.Z., Theodoulidis, B.: Analyzing stock market fraud cases using a linguistics-based text mining approach. In: WaSABi-FEOSW@ ESWC (2014)
16. Schumaker, R.P., Chen, H.: Textual analysis of stock market prediction using breaking financial news: the AZFin text system. ACM Trans. Inf. Syst. (TOIS) **27**(2), 12 (2009)
17. Sorto, M., Aasheim, C., Wimmer, H.: Feeling the stock market: a study in the prediction of financial markets based on news sentiment. In: 2017 Proceedings of the Southern Association for Information Systems Conference, St. Simons Island, GA, USA (2017)
18. Khedr, A.E., Salama, S.E., Yaseen, N.: Predicting stock market behavior using data mining technique and news sentiment analysis. Int. J. Intell. Syst. Appl. (IJISA) **9**(7), 22–30 (2017)
19. Tasci, S., Gungor, T.: An evaluation of existing and new feature selection metrics in text categorization. In: 23rd International Symposium on Computer and Information Sciences (ISCIS), pp. 1–6. IEEE (2008)
20. Forman, G.: An extensive empirical study of feature selection metrics for text classification. J. Mach. Learn. Res. **3**(Mar), 1289–1305 (2003)
21. Martinez-Romo, J., Araujo, L.: Detecting malicious tweets in trending topics using a statistical analysis of language. Expert Syst. Appl. **40**(8), 2992–3000 (2013)
22. Kaya, M.Y., Karsligil, M.E.: Stock price prediction using financial news articles. In: 2nd IEEE International Conference on Information and Financial Engineering (ICIFE), pp. 478–482. IEEE (2010)
23. Myung, J., Yang, J.Y., Lee, S.G.: Picachoo: a tool for customizable feature extraction utilizing characteristics of textual data. In: Proceedings of the 3rd International Conference on Ubiquitous Information Management and Communication, pp. 650–655. ACM (2009)

24. Liaw, A., Wiener, M.: Classification and regression by randomForest. R News **2**(3), 18–22 (2002)
25. Kotsiantis, S.B., Zaharakis, I., Pintelas, P.: Supervised machine learning: a review of classification techniques. Informatica **31**, 249–268 (2007)
26. Bradley, P.S., Fayyad, U.M., Reina, C.A.: Clustering very large databases using EM mixture models. In: 2000 15th International Conference on Pattern Recognition, vol. 2, pp. 76–80. IEEE (2000)

Stock Market Real Time Recommender Model Using Apache Spark Framework

Mostafa Mohamed Seif(✉), Essam M. Ramzy Hamed(✉), and Abd El Fatah Abdel Ghfar Hegazy(✉)

Arab Academy for Science, Technology and Maritime Transport, Cairo, Egypt
mmseif87@yahoo.com, {essam.hamed,Ahegazy}@aast.edu

Abstract. The stock market is considered a complicated and nonlinear system. Now stock market prediction is recognized as an attracting point for financial investors. The historical price is not considered as the main factor to predict the stock market trend. There are many other factors such as politics and natural events that affect social media environments like Twitter and Facebook which generate huge datasets needed data analysis to extract the polarity of these data and its effectiveness on the stock market. On the other hand, these data may be unstructured and need special handling on storing and processing. This paper proposes a real-time forecasting of stock market trends based on news, tweets, and historical price. A supervised machine learning algorithms used to build this model. Historical price will be combined with sentiment analysis to build the hybrid model based on Apache Spark and Hadoop HDFS to handle big data (structured and unstructured) generated from social media and news websites. The proposed model works in two modes; the offline mode that works on historical data including today's data after ending of a stock market session, and real-time mode that works on real-time data during the stock market session. This model increases the accuracy of prediction due to the additional features added by sentiment analysis on StockTwits and market news data. In addition, this model enhances the performance of handling this data set due to parallel processing occurred on data using Apache Spark.

Keywords: Sentiment analysis · Supervised learning · Apache Spark · Big data
Hadoop · HDFS · StockTwits

1 Introduction

The stock market prediction has become a very important topic nowadays as it is used by business people. There are two traditional methods used to predict stock market fundamental analysis and technical analysis. Fundamental is a technique used to evaluate the security by studying everything that can affect the security's value, including economic influences (like the overall economy and industry conditions) and individually influences (like the management of companies and financial condition). A fundamental analysis looks like the balance sheet, loss statement, the profit, financial ratios and other data that could be used to predict the future of a company [1–3].

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 671–683, 2018.
https://doi.org/10.1007/978-3-319-74690-6_66

The main disadvantage of fundamental analysis technique is time-consuming, it can also get you on board a good stock but at the wrong time and you may need to hold on to the stock for a long time. Technical analysis has nothing to do with the financial performance of the underlying company. In this method, the analyst simply studies the trend in the share prices. The underlying assumption is that market prices are a function of the supply and demand for the stock, which, in turn, reflects the value of the company. This method also believes that historical price trends are an elevator of the future performance. There are some drawbacks in using technical analysis as well. They are as difficult to master and check their accuracy and validation against a biased view. The main issues on these two methods are not considered as the mirror to events that happen in any country during market session, so this paper introduces another prediction model to enhance the accuracy of prediction and reflect real-time market events and their effect on prices. The model uses sentiment analysis to analyze the data generated from news websites and social media using supervised machine learning algorithms and combine the results with historical price as an additional feature to find the final classification result as binary classification up or down. On the other hand, some well-known companies like Google, Amazon, LinkedIn, and Yahoo have generated a huge amount of structured and unstructured data every day that needs huge storage to store these big data and high-performance processing environments to process it in few minutes.

The problem is how to enhance the accuracy of stock market prediction using real-time data generated from sources like social media and news channels, store it on data storage, and make parallel processing on it to enhance the recommendation delivery time to the stock market trader. The proposed model built on Hadoop Distributed File System (HDFS) as data storage and use Apache Spark framework on data processing using Resilient Distributed Dataset (RDD) to parallelize data processing, make the best utilization of resources, and get quick prediction results.

1.1 Literature Review

In the stock market, you have to take the right decision on the right time to gain profit and maximize your wealth. This right decision will be through buying or selling a stock and the decision taken depends on many factors. Nowadays most of the stock analytics predict stock prices depending on historical data, but with the huge amount of data today and data variety, it must use new data sources to increase the accuracy of prediction and find new ways to take a right decision. However, this takes place through handling all types of data at the same time, with these huge amounts of data need a new approach and brilliant data processing framework.

Nayak et al. [4] used the neural network to predict stock market price using Hadoop and MapReduce by building two models one for daily prediction based on the combination between historical price and tweets sentiment analysis and the accuracy was up to 70%. The second model finds the stock market trend correlation between two months and the correlation result was very small. Mukesh and Rohini [5] used a neural network and Hadoop HDFS to compare between two algorithms and proved that Least Square Algorithm better than Sigmoid Algorithm by calculating Root Mean Square "RMS" error. He also proved that using Hadoop MapReduce and Hadoop HDFS in parallel processing is better than using single node processing. Bachhav et al. [6] presented a method to make sentiment analysis using machine learning techniques and Hadoop HDFS as data storage and analyze online feedback of users from online sites to detect impressions and make sentiment analysis of a specific topic. Khairnar and Kinikar [7] used Support Vector Machine (SVM) - a machine learning technique - and Hadoop HDFS to prove that the accuracy of sentiment analysis using Latent Semantic Analysis "LSA" is more accurate than using SVM only and that it enhances the processing of data by using Hadoop MapReduce. Ghaiehchopogh et al. [8] applied linear regression algorithms using Relational Database Management System (RDBMS) to calculate the relation between two variables "average price and volume" per day to predict the next stock market price after comparison occurred between results observed and stock market values, he obtained a similarity of 61.35%.

The remainder of this paper is structured as follows: Sect. 2, presents the proposed stock market real time recommended model. Section 3, presents experimental results. Section 4, discusses conclusion.

2 The Proposed Stock Market Real Time Recommended Model

The proposed model is designed based on three main phases: data acquisition phase, data storage phase and data analysis phase as shown in Fig. 1. Apache Spark [9, 10] is used to handle these phases. It is the newer framework built on the same concepts and techniques of Hadoop. However, Hadoop is the best solution for large data processing; it drops on some scenarios especially on iterative algorithms. Another problem on Hadoop is that it does not cache intermediate data for faster performance. It releases the data to the disk between each step. In contrast, Spark uses RDD to persist the data on the worker's memory and the concept of caching to avoid reproducing all the pipeline processes when the task is failed. Spark applications run as isolated sets of processes on a cluster coordinated by the SparkContext object in the main program (the driver program). Specifically, to run on a cluster, the SparkContext connects with Cluster Manager that allocates resources across applications. As shown in Fig. 2, when the SparkContext is connected, it acquires executors on nodes in the cluster, which run computations and store data for the application. Then, it sends the application code to the executors. Finally, SparkContext sends tasks for the executors to run.

On data acquisition phase, the used dataset consists of three sources: StockTwits, Market News, and Historical Prices. They have been collected in the interval from the period 1/2/2013 to 30/6/2016. StockTwits is used as the source of social media data. Its content is focused on the discussion about stock markets. It is believed that the user on StockTwits has good experience to write tweets related to stock markets and financial topics. StockTwits creates $Ticker tag to enable and organize "Streams" of information around stocks and markets across the web and social media. Every tweet includes information about creation date, message content and message source.

StockTwits is collected for three companies Apple ($AAPL), International Business Machines ($IBM) and Google ($GOOG). Market news is used to reflect the pulse of the market and mirror the events occurring on the market during a stock market session.

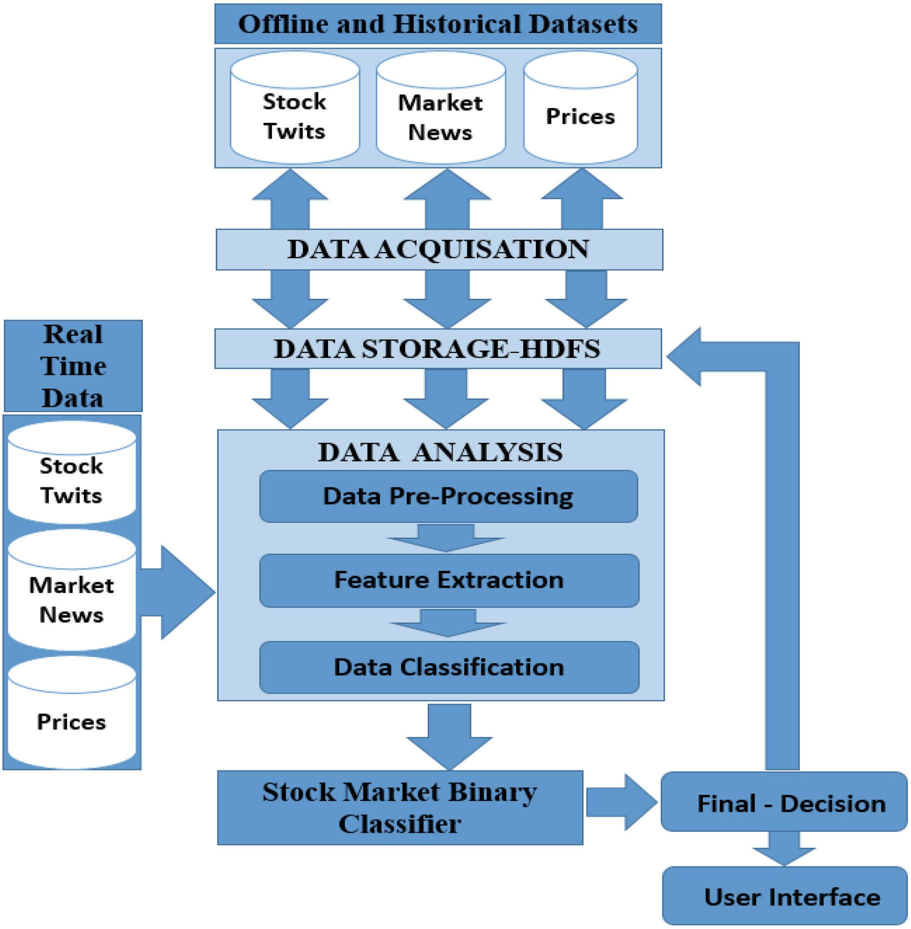


Fig. 1. The proposed framework

Examples of market news are politics news and public events. Historical prices are used as the main data source of stock market prediction algorithms. Yahoo finance is used to get data related to $AAPL, $IBM and $GOOG stocks and retrieve data related to stock prices details.

The proposed model works with two modes: real-time mode and offline mode. In real time mode, the model is running only on real-time data. After creation of both StockTwits or market news, an event fired and the model triggered to work and classify tweet or news body to get its polarity positive or negative then combines the result of classification with price of this stock at this moment to give final recommendation to trader, so on this model the main features needed are like open price, high price and low price from historical stock prices data, and from tweets and market news data the message body and date are only taken. Offline mode works after a stock market session is ended because new features are already generated like close price, adjusted close

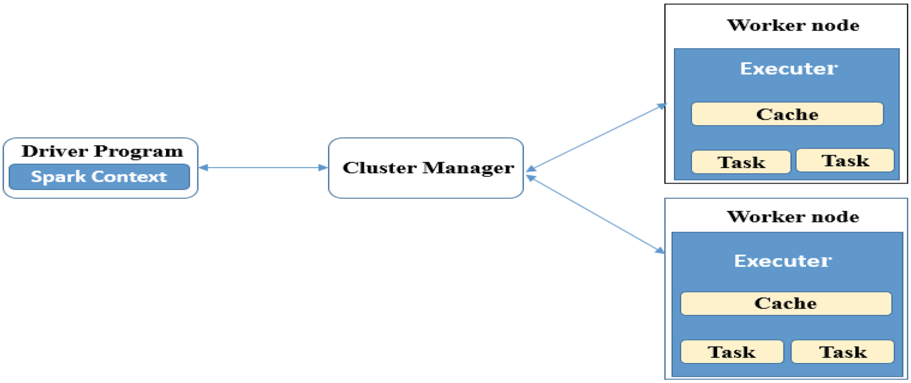


Fig. 2. Spark architecture

price- AdjClose- (which contains close price with considerations to any dividends occur on this stock) and total volumes plus extra features will be generated from the main features that already exist like the change that occurs on close price between current close price and number of days. On the other hand, the accumulative sentiment analysis of tweets and news generated during this day will be considered as features in offline mode. All these features will be explained on feature extraction section.

On data storage phase, HDFS is used to store data collected from multiple data sources. HDFS is the file system component of the Hadoop framework. HDFS is designed to play down the storage overhead and mine a large amount of data on distributed fashion hardware. Every file stored on HDFS is divided into 128 MB with three copies stored on three different nodes on Hadoop cluster. As shown in Fig. 3, the cluster has two main components: Name Node and Slave Nodes. Name Node contains the

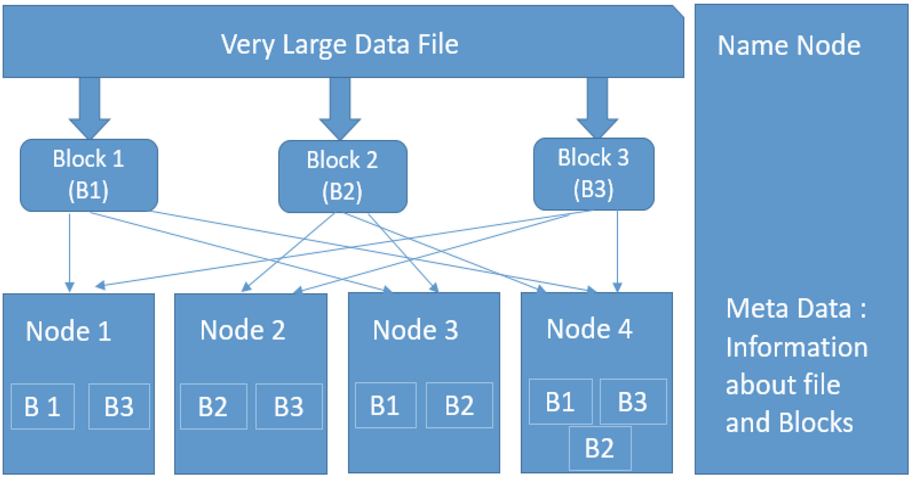


Fig. 3. HDFS architecture

Metadata file which contains the location of each block. Slave Nodes contains the data themselves [11, 12].

In data analysis phase, the meaningful knowledge is extracted from data stored. A set of RDD transformations and actions are implemented on a dataset to make data pre-processing. Spark offers many machine-learning algorithms already implemented on MLib which is Apache Spark scalable machine learning library and it is developed as part of Apache Spark Framework. It contains many implemented machine learning techniques such as classification, clustering, and regression. Spark offers many Application Programming Interface (APIs) in Scala, R and Python languages which are run on HDFS. The python's API is used to build this model. The analysis phase consists of three steps: data pre-processing, feature extraction and data classification.

Data Pre-processing: Before carrying out any "mining" activities, text in StockTwits and news needs to be prepared or pre-processed in a way that can enable mining algorithms to be applied to it. There are many techniques used to pre-process the data before using it, like Tokenization, Case Folding, Lemmatization, and Stemming.

Feature Extraction: It involves reducing the number of resources required to describe a large set of data. On StockTwits the features used are the message content, message source, message time and cash tag of a tweet. For market news, the features used are news content and date. On stock prices, the features used in offline mode are the close price, AdjClose, volume but low price, open price, and high price are used on both offline and online modes. Additional features have been extracted from data already existing only on offline mode. The list of this additional features added is as follows:

Days Return: Percentage difference of adjusted close price of i-th day and (i − 1)th day.

$$\mathrm{Return(i) = AdjClose(i) - AdjClose(i-1)/AdjClose(i-1)} \tag{1}$$

Where,

- *Return(i)* is the change occurred from one day ago.
- *AdjClose(i)* is the adjusted close price today.
- *AdjClose(i − 1)* is the adjusted close price yesterday.

Multiple Day Returns: Percentage difference of AdjClose Price of i-th day compared to (i-delta)th day. *Example*: 2-days Return is the percentage difference of AdjClose price of today compared to the one of two days ago.

$$\mathrm{Return(n) = AdjClose(i) - AdjClose(i-n)/AdjClose(i-n)} \tag{2}$$

Where,

- (*n*) is the number of days.
- *Return(n)* is the change occurred from n days ago.
- *AdjClose(i)* is the adjusted close price today.
- *AdjClose(i−n)* is the adjusted close price on days within n days.

Returns Moving Average: Average returns on last delta days. Example: 2-days Return is the percentage difference of Adjusted Close Price of today compared to the one of 2 days ago.

$$\text{MovAvg(n)} = \text{Return1} + \text{Return2}/\text{Return(n)} \tag{3}$$

Where,

- (n) is the number of days.
- *MovAvg(n)* is the Returns average of n days.

Data Classification: For StockTwits and Market News, Sentiment analysis is used to analyze data and it is the main method to get the polarity of human opinion from comments they write. Machine learning algorithms are used to implement sentiment analysis on social media data, Naïve Bayes classifier used to implement classification of StockTwits and market news datasets. Naïve Bayes is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naïve Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. Naïve Bayes model is easy to build and particularly is useful for very large datasets. Along with its simplicity, Naive Bayes is known for outperforming even with highly sophisticated classification methods. Bayes theorem provides a way of calculating posterior probability P(c|a) from P(c), P(a) and P(a|c). For historical stock prices combined with the results extracted from sentiment analysis classifiers, machine learning supervised classification techniques are used like Random Forest, Support Vector Machine, and Logistic Regression.

$$\text{P(c|a)} = \text{P(a|c)}\,\text{P(c)}/\text{P(a)} \tag{4}$$

Where,

- *P(c|a)* is the posterior probability of class (c, target) given predictor (a, attributes).
- *P(c)* is the prior probability of class.
- *P(a|c)* is the probability of predictor given class.
- *P(a)* is the prior probability of predictor.

3 Experimental Results

For testing and training data, two datasets are used, one for the real-time mode that contains real-time features and the other contains features of offline mode. There are three channels of data provided on this dataset. The main channel is the historical prices. Yahoo finance website is used as the provider of stock prices of $AAPL, $IBM, and $GOOG [13]. StockTwits website is used as the channel to collect tweets for the stock market [14]. The last channel used to fetch market news is Reddit world channel which contains historical news headlines [15].

All data crawled on date interval from 1-2-2013 to 30-6-2016. The dataset is divided into 80% for training and 20% for testing. Weka tool is used to test proposed model.

Weka is an open source tool used in data mining that contains many already-implemented machine learning algorithms. The dataset of $AAPL is classified on Weka using multiple classifier algorithms like Naïve Bayes (NB), Logistic Regression (Log-Reg), Decision Tree (DT), K-Nearest Neighbor (KNN), Support Vector Machine (SVM) and Random Forest (RF) to choose the best three of them and after run, RF, Log-Reg, and SVM have been chosen as the best algorithms as shown in Fig. 4. Requests python package is used to stream data from StockTwits and Reddit world channel. On the other hand, Yahoo Finance Python package is used to fetch data of current prices for a specific stock. Two sentiment analysis classifiers have been built, one of them to classify any new StockTwits and the other for market news. The results extracted from two classifiers combined with stock price data that entered to another classifier to get final recommendation.

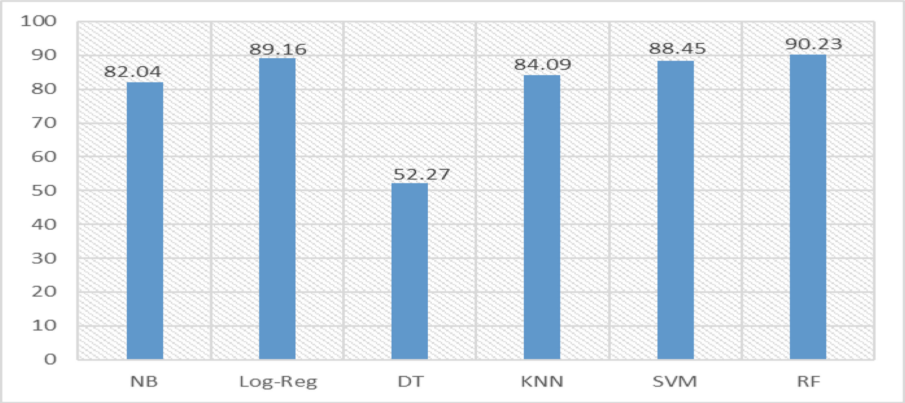


Fig. 4. Accuracy of data mining algorithms

The proposed model trained on an offline model using offline mode dataset and trained in online mode using real-time mode dataset. The model works on offline mode after the stock market session ended and fetch the complete stock prices data that contains close prices and the total volume of the day. The model works on real-time mode during market session time. The model trained by fetching data from sources and storing it on HDFS then transferring it to analyze through three predefined phases to be ready as training data for sentiment analysis classifier and the results combined with stock prices to enter as the data source into final classification phase to give next day recommendation. The same steps will be implemented on real-time model using real-time data sets but do not contain any close price or volumes data because during stock market session this data are still unknown.

After these phases are finished, two Stock market binary classifiers are generated, one for offline mode and the other for a real-time mode. The results of the proposed model are compared with weka tool results. Figure 5 constructs the comparison between Data mining techniques implemented on weka tool and proposed model using offline dataset without adding sentiment analysis features. Figure 6 shows the same comparison results on the offline dataset with sentiment analysis results. Figure 7 uses real-time dataset on comparison without sentiment analysis features, and Fig. 8 makes the last comparison using sentiment analysis features.

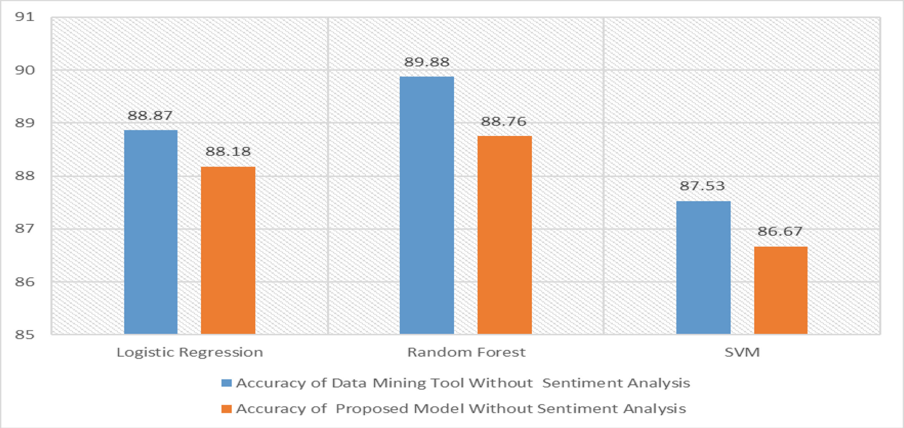


Fig. 5. Accuracy results of data mining techniques vs proposed model without sentiment analysis features (offline dataset)

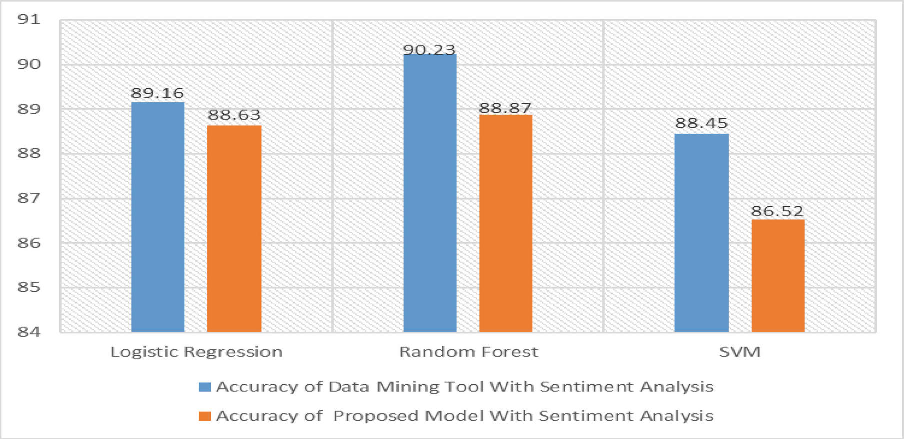


Fig. 6. Accuracy results of data mining techniques vs proposed model with sentiment analysis features (offline dataset)

The real-time proposed model was run on 22/9/2017 from 12:08 PM to 12:21 PM and for every new tweet created on StockTwits or market news published on Reddit World Channel, the proposed model triggered to implement algorithm by preprocessing tweet or news and makes sentiment analysis to calculate the polarity of tweet or market news. The model then fetches the current open price, low price, and high price and combines it with sentiment analysis result to compose the feature vector. The proposed model takes this feature vector and implements the prediction and the final result is binary result “1” or “0”. Value “1” represents the recommendation as buying and “0” represent recommendation as they sell. Figure 9 shows the recommendations given to trader on three companies Apple, IBM and Google.

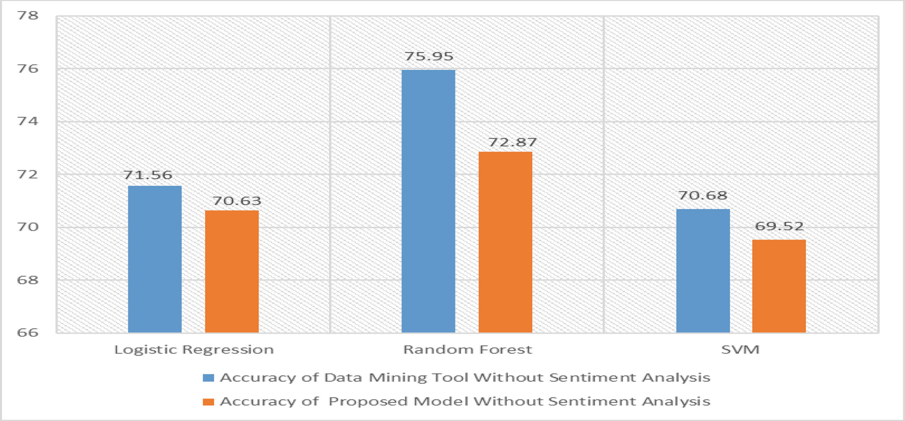


Fig. 7. Accuracy results of data mining techniques vs proposed model without sentiment analysis features (real-time dataset)

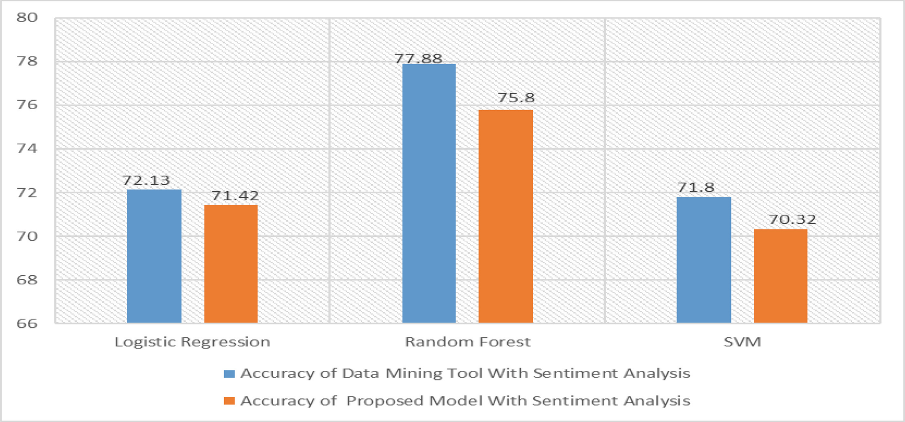


Fig. 8. Accuracy results of data mining techniques vs proposed model with sentiment analysis features (real-time dataset)

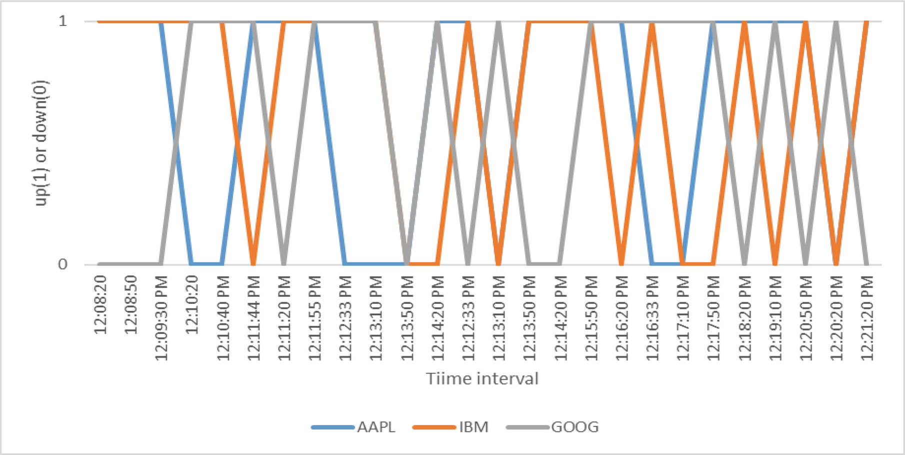


Fig. 9. Model results on three Stock AAPLE, IBM and Google

This case study ran on spark cluster installed on Amazon EC2. According to the dataset used in this case. The time consumed to train model to generate market news sentiment analysis classifier, StockTwits classifier, and the final stock market binary classifier as shown in Fig. 10.

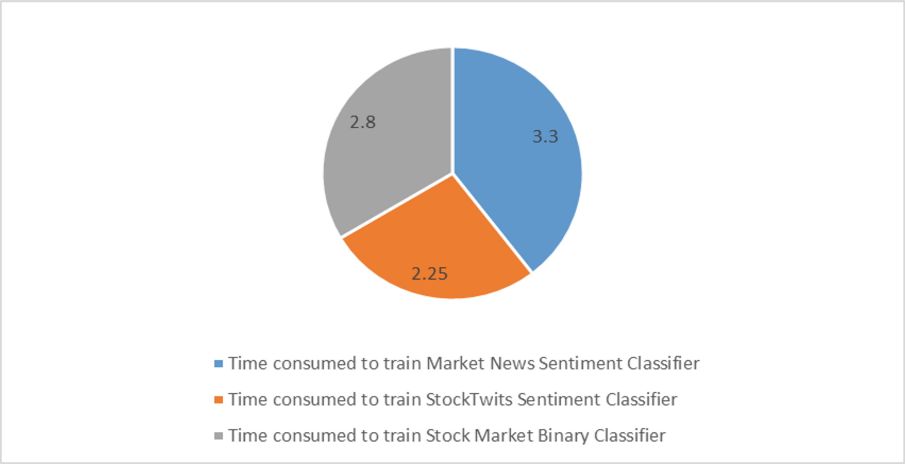


Fig. 10. Spark processing time

4 Conclusion

This work presents a model to predict stock market trend and gives the recommendation to the trader using the combination between stock price and sentiment analysis on social media data and market news under big data environment. Three types of the dataset used historical stock prices, StockTwits and market news are fetched from multiple data sources and stored in HDFS. Sentiment analyzer has been built to analyze StockTwits and market news data, then the features extracted from them are combined with stock price features and compose another dataset used to build our new classifier. Our model works on two modes, offline mode, and real-time mode. The offline mode works on end of day data like close price, AdjClose price, volume, the accumulative sentiment analysis of all twits and news during the day in addition to normal stock price features. On the other hand, the real-time mode works on live features like open price, high price, low price plus fresh tweets and news generated during the stock market session. All the classified algorithms are implemented with Apache Spark using Apache Spark machine learning libraries, entitled MLib to enhance the performance of processing. The result extracted from Weka tool is compared with the result observed from proposed model and it seems to be more relevant to it.

References

1. Moosa, I., Li, L.: Technical and fundamental trading in the Chinese stock market: evidence-based on time-series and panel data. Emerg. Mark. Financ. Trade **47**(1), 23–31 (2011)
2. Venkatesh, C.K., Tyagi, M.: Fundamental analysis as a method of share valuation comparison with technical analysis. Bangladesh Res. Publ. J. **5**(3), 167–174 (2011)
3. Drakopoulou, V.: A review of fundamental and technical stock analysis techniques. J. Stock Forex Trad. **5**, 163 (2015). https://doi.org/10.4172/2168-9458.1000163
4. Nayak, A., Pai, M.M.M., Pai, R.M.: Prediction models for indian stock market. In: Twelfth International Multi-Conference on Information Processing-2016 (IMCIP-2016), Procedia Computer Science, vol. 89, pp. 441–449 (2016)
5. Mukesh, Rohini, T.V.: Market price prediction based on neural network using hadoop mapreduce technique. In: Computational Systems for Health & Sustainability
6. Bachhav, C., Gite, M., Jadav, K., Malode, K.: Sentimental analysis on big data. Int. J. Res. Eng. Appl. Manag. (IJREAM) **1**(1) (2015)
7. Khairnar, J., Kinikar, M.: Sentiment analysis based mining and summarizing using SVM-MapReduce. Int. J. Comput. Sci. Inf. Technol. (IJCSIT) **5**(3), 4081–4085 (2014)
8. Ghaiehchopogh, F.S., Bonaband, T.H., Rezakhaze, S.: Linear regression approach to prediction of stock market trading volume: a case study. Int. J. Manag. Value Supply Chains (IJARCE) **4**(3), 25–31 (2013)
9. Sasmita, P., Lenka, R.K., Stitipragyan, A.: A hybrid distributed collaborative filtering recommender engine using apache spark. Procedia Comput. Sci. **83**, 1000–1006 (2016)
10. Etaiwi, W., Biltawi, M., Naymat, G.: Evaluation of classification algorithms for banking customer's behavior under apache spark data processing system. Procedia Comput. Sci. **113**, 559–564 (2017)
11. Gerard: Hadoop Essentials – The Eight Things You Need To Know. Working Analytics (2015). http://workinganalytics.com/hadoop-essentials-by-cloudera-eight-lessons-learned/. Accessed 30 Sep 2017
12. White, T.: The Hadoop distributed file system. In: Hadoop: The Definitive Guide, pp. 45–80. O'Reilly&Associates, Sebastopol (2012)

13. Yahoo Finance: Historical Price. https://finance.yahoo.com/lookup. Accessed 30 Aug 2017
14. StockTwits: https://stocktwits.com/home#people-and-stocks. Accessed 30 Aug 2017
15. RichardChen: Sentiment effect on stock price. https://www.kaggle.com/otwordsne1/sentiment-effect-on-stock-price-draft/data. Accessed 30 Aug 2017

Patient Fingerprint Minutiae Based Medical Image Watermarking and Adaptive Integrity

Lamri Laouamer[1,2](✉) and Meryam El Mouhtadi[3]

[1] Department of Management Information Systems, CBE, Qassim University,
Buraydah, Kingdom of Saudi Arabia
laoamr@qu.edu.sa

[2] Lab-STICC (UMR CNRS 6285), European University of Bretagne,
Brest, France

[3] LRIT – CNRST URAC29, Faculty of Science,
University Mohammed V, Rabat, Morocco
meryem.mouhtadi@gmail.com

Abstract. The robustness and integrity of multimedia content is an important issue in data security. Digital watermarking can contribute significantly to verify both robustness and integrity and to prove any illegal manipulation. Through this paper, we propose a new watermarking approach verifying the integrity of the patient's medical images contents. The proposed approach is based on extracting the minutiae from the patient's fingerprints to serve as a proof of integrity. Choosing minutiae was also aimed to reduce the computational complexity in the watermark embedding and extracting processes to be well adapted for the real-time telemedicine applications compared to the existing works. We tested our approach against the most common attacks to judge the performance of the proposed approach. The obtained results are very encouraging in which we detail through this paper.

Keywords: Robustness · Integrity · Attacks · Medical image
Fingerprint · Minutiae

1 Introduction

For the security of telemedicine applications, where the developed methods and protection tools are inadequate, this can create a lot of worries on credibility of the information exchanged through patient's circuit, medical institutions and insurance companies. While most IT professionals are well aware of this, it is clear that few companies are willing to make the necessary investments to implement an adaptable and scalable solution with a minimum cost and less complexity in terms of computation. When traditional defenses are no longer sufficient to cope with the diversity and volume of threats where the information system is exposed, effective and efficient solutions become a primary necessity.

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 684–692, 2018.
https://doi.org/10.1007/978-3-319-74690-6_67

The integrity concept is an important parameter in information security. Its definition is based on a decision which guarantees that the received data are strictly identical to those emitted. It is preferable to hide the concerned data for authentication in the image itself in the form of watermark, rather than in a separate file as in the case of an external signature [1].

Digital watermarking can enhance security in favor of the image data reliability by embedding in the host image an informed signature inspired from the host image (proof of integrity). It can also be used to integrate confidential data such as physiological (ECG …) and/or diagnostic information, biometric evidence. This ensures greater confidentiality of patient data. The information reliability should be involved as the combination of a double verification: data integrity and authenticity. The image must not have voluntary and/or involuntary modifications since its acquisition or after any treatment forming part of a fully defined and known protocol. The image should also be in line with the patient's identity (authenticity).

The authors in [2] proposed their approach for securing medical images based on using a dual watermarking method. The first watermark is considered as robust to reach a high degree of similarity between the embedded watermark and the extracted one after attacks. The second watermark is fragile whose role is to guarantee the integrity of the exchanged medical images. They considered the edge information when the embedding locations for the fragile watermarking were selected. And ROI was explored to the significant zones on medical images. This approach can guarantee that the embedded watermarks are detected accurately without any interference. This approach requires a huge number of calculations and it would be difficult to adapt it for a real time applications.

In [3] a digital blind forensics approach for medical imaging has been proposed to check the integrity against some kind of attacks such filtering, loss compression, etc. The proposed approach is based on comparing the histogram statistics of the DC transform coefficients and the histogram statistics of the Tchebichef moments. Both features serve to be an input of a set of SVM classifiers to eliminate tampered images from original and identifying the global modification one image. Results evaluation shows that these image features can help to blindly distinguish modifications with a detection rate greater than 70%.

In another hand, authors in [4] presented a medical image watermarking technique by introducing a watermark represented by the patient identity in singular values of the DWT coefficients (one level) with minimal distortion on medical information. The proposed framework has been reached with good imperceptibility and improved security. The watermark does not get disrupted when the watermarked image is altered. The obtained results in terms of PSNR and Correlation coefficients between the original watermark and the extracted one against each attack were very encouraging. The main loophole is that the proposed method needs a high complexity in term of computation.

We present in this paper a new watermarking approach for verifying the integrity of medical images transmitted through communication networks based on the most relevant extracted minutiae of the patient's fingerprint that will be considered as integrity evidence. This choice allow to build an informed watermark based on the most relevant minutiae such ridges and bifurcation by reducing significantly the complexity when achieving watermark embedding/extraction processes as well as the integrity verification to be well adapted for a real time applications.

2 Preliminaries

Fingerprints also called dermatoglyphs are a signature that we leave behind each time we touch an object. The patterns drawn by the ridges and folds of the skin are different for each individual; this is what motivates their use by the criminal police for several years. The probability of finding two similar fingerprints is 1 of 10^{24}. Twins, for example, coming from the same cell, will have very close and not similar fingerprints [5]. Fingerprints are classified according to a decade-old the Henry system [6, 7]. The classification is based on the general topography of the fingerprint.

We list 13 different types of minutiae that can be used to classify fingerprints and to ensure their uniqueness, the 6 most common are presented in Fig. 1. Since ridges and bifurcations are the two main minutiae that successfully characterize a fingerprint. It allows considering them as a good choice to build the integrity evidence vector in our watermarking model.

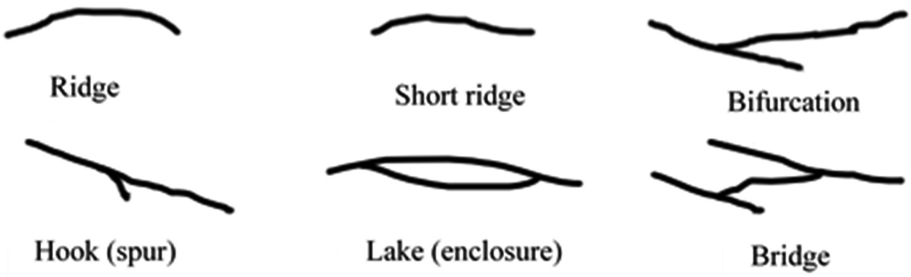


Fig. 1. The six main types of minutiae

3 System Model

The model we propose consists of a medical images watermarking framework operating in the spatial domain by reaching also integrity of the exchanged medical images through a communication channel. The watermark will be embedded directly in the pixels values of host image without applying any transform. The embedding process will be done on image region of interest ROI of course by preserving a visual quality almost similar to the original image called imperceptibility. The watermark embedding algorithm involves hiding the fingerprint watermark. The watermark visibility/invisibility can be carried out by a parametrization of the linear interpolation factor β given by the Eq. (1)

$$i_w = (1-\beta)w + \beta \times i, \quad 0 < \beta < 1 \tag{1}$$

Where: i, w, i_w are respectively the host image, watermark and the watermarked image. Controlling the visibility/invisibility of the watermark can be expressed as follows:

$$\xrightarrow{\beta} 0 \underset{visibility}{\Rightarrow} i_w = (1-\beta)w + \beta \times i \rightarrow w$$

$$\xrightarrow{\beta} 1 \underset{invisibility}{\Rightarrow} i_w = (1-\beta)w + \beta \times i \rightarrow i$$

3.1 Fingerprint Preprocessing

After extracting the two main minutiae (ridge and bifurcation), there are a certain number of minutiae which can be considered as false generated in the image contour and which will be eliminated by taking into account only the fingerprint ROI. The binarization and squelatization steps allow respectively removing the pixel background and reducing the thickness of lines in pixel. For this purpose, we used the Otsu [8] (Fig. 2) method to perform binarization. As well as the Zhang [9] method in the case of squeletization (Fig. 2).

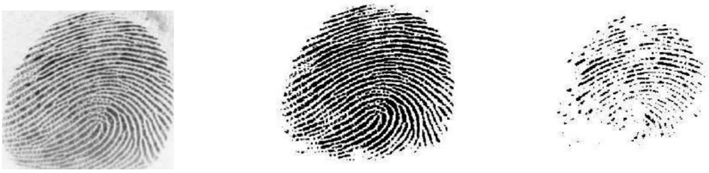


Fig. 2. (a) Original fingerprint, (b) Fingerprint binarization with Otsu approach, (c) fingerprint squeletization with Zhang approach.

3.2 Watermark Embedding and Extraction

Figure 3 clearly illustrates the proposed system model from fingerprint extracting minutiae to extracting the attacked watermark w_a after applying attacks on the watermarked image iw_a. The system is divided into three main stages. The first stage consists to the fingerprint preprocessing which aim to binarize and skeletonize the fingerprint and extracting the main two minutiae such ridge and bifurcation. After extracting these two minutiae, we proceed to the watermark (second stage) based on fingerprint. The embedding watermark within the original image will be conducted by a linear interpolation of type:

$$i_w = (1 - \beta)w + \beta \times i, \quad 0 < \beta < 1 \tag{2}$$

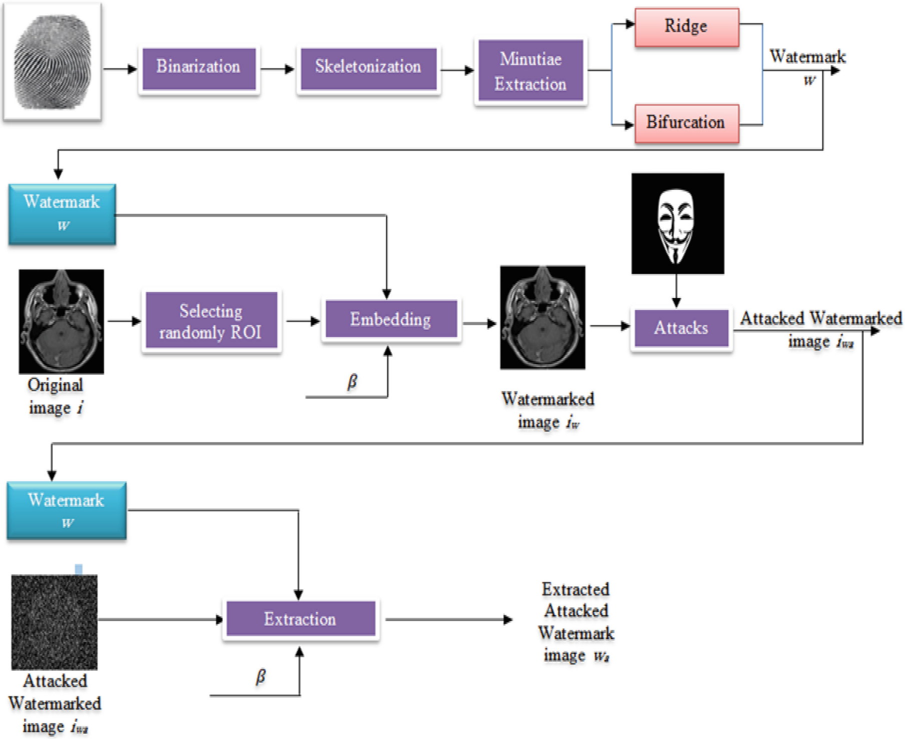


Fig. 3. Proposed system model for robustness and integrity verification.

This step will be reached on pixels of the ROI medical image. In the performed tests, we chose the value of β close to 1 to achieve the watermark imperceptibility. The last stage consist to extract the attacked watermak w_a after applying some known geometric/non-geometric attacks through Stirmark Benchmark [10]. The watermark extraction is given by the following formula:

$$w_a = \frac{1}{t} w - \frac{1-t}{t} I_{wa}, \quad 0 < t < 1 \tag{3}$$

To check the integrity of the exchanged watermarked image, in the receiver side we should compare the received minutiae with the original one. If we notice that there are changes in this vector, we conclude that the integrity cannot be reached and there is some falsification on the watermarked image.

3.3 Results Evaluation

We have tested the performance of our proposed watermarking model only in the case of an invisible watermarking since it consists to achieve a reasonable robustness which is one of the paper objectives. This means that the used linear interpolation parameter is close to 1, → 1. Similarly, the applied attacks against the watermarked images are performed by Stirmark Benchmark [10].

$$PSNR = 10log_{10}\left(\frac{255^2}{\frac{1}{\mathrm{M}\times\mathrm{N}}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(X_{ij} - X'_{ij}\right)^2}\right)\mathrm{dB} \tag{3}$$

$$CC = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{\left(\sum_m \sum_n (A_{mn} - \bar{A})^2\right)\left(\sum_m \sum_n (B_{mn} - \bar{B})^2\right)}} \tag{4}$$

To decide on the robustness effectiveness of our proposed system, we used the most popular metrics in watermarking field as illustrated in Table 1.

Table 1. Used attacks description

Attack name	Description
ROT_45	Rotation with 45°
JPEG_90	JPEG compression with quality factor = 90
MEDIAN_9	Median filtering (3 × 3)
RML_30	Remove lines (30)
NOISE_80	Noise with density degree = 80

These metrics consist of calculating the Peak Signal Noise Ratio PSNR and the degree of CC correlation [11] between the original watermark w and the extracted one w_a against each attack. These two metrics are well defined respectively in Eqs. 3 and 4. The achieved results are well illustrated in Figs. 4 and 5. In the most of cases, we found the PSNR values exceed the 34 db and the CC coefficients are very close to 1, which means that the robustness is almost well reached instead the proposed framework is operated in spatial domain.

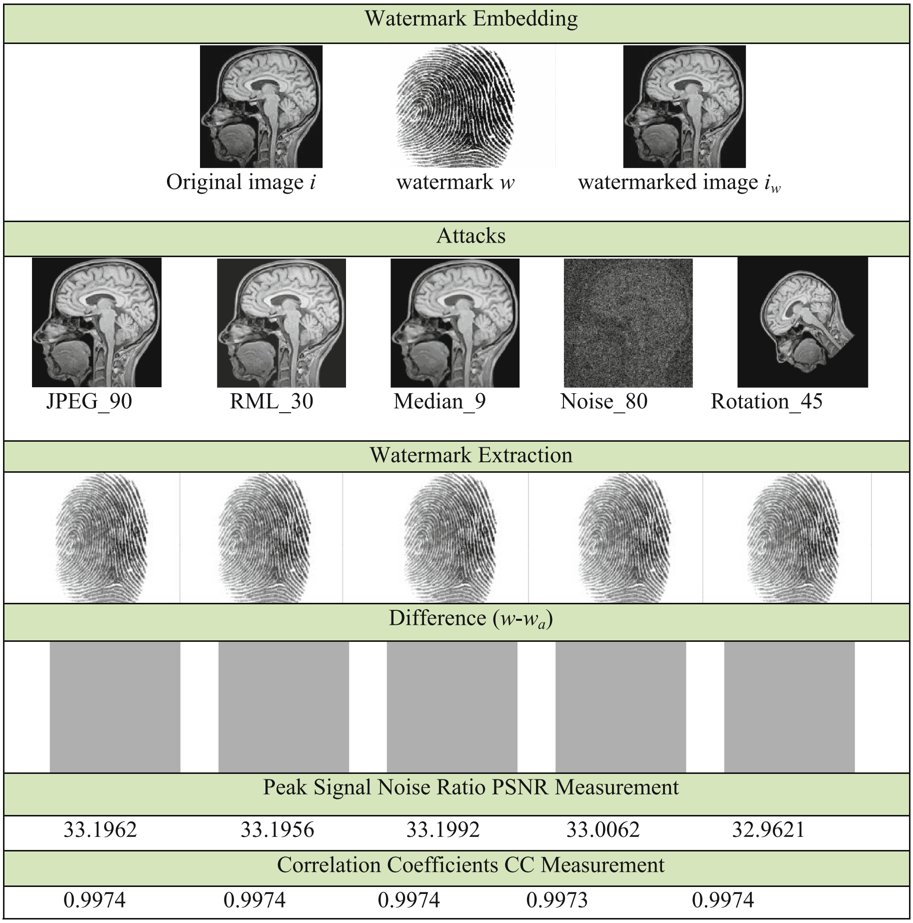


Fig. 4. The achieved results in the case of head image

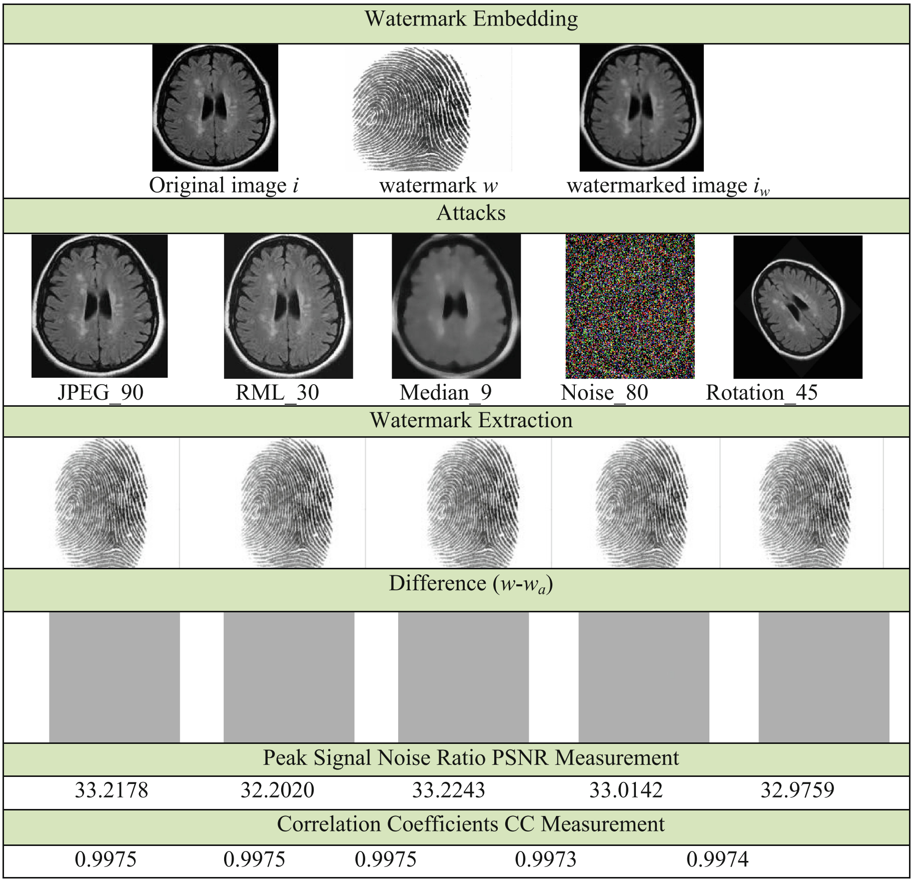


Fig. 5. The achieved results in the case of brain image

4 Conclusions

The proposed watermarking framework in this paper consists of reaching watermark reasonable robustness against the most known attacks either geometrical or non-geometric. As well as the integrity assurance of the watermarked images exchanged through communication networks. The Integrity factor is based on the extraction of the most important minutiae in fingerprint such ridge and bifurcation. This process makes possible checking the extracted minutiae before emission and those at the receiver side in order to asses on verification of the medical image owner. This process can be achieved at the receiver level through one of our previous work. The obtained results are very encouraging and allow us to extend our tests on a large database of medical images and fingerprints.

References

1. Pasquini, C., Brunetta, C., Vinci, A.F., Conotter, V., Boato, G.: Towards the verification of image integrity in online news. In: 2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW) pp. 1–6 (2015)
2. Lim, S.J., Moon, H.M., Chae, S.H., Pan, S.B., Chung, Y., Chang, M.H.: Dual watermarking method for integrity of medical images. In: 2008 Second International Conference on Future Generation Communication and Networking, Hainan Island, China, pp. 70–73 (2008)
3. Huang, H., Coatrieux, G., Shu, H., Luo, L., Roux, C.: Blind integrity verification of medical images. IEEE Trans. Inform. Technol. Biomed. **16**(6), 1122–1126 (2012)
4. Singh, A., Nigam, J., Thakur, R., Gupta, R., Kumar, A.: Wavelet based robust watermarking technique for integrity control in medical images. In: 2016 International Conference on Micro-Electronics and Telecommunication Engineering, Ghaziabad, India, pp. 222–227 (2016)
5. Elmouhtadi, M., Elfkihi, S., Aboutajdine, D.: Fingerprint identification using hierarchical matching and topological structures. In: 2nd International Conference on Advanced Intelligent Systems and Informatics AISI 2016, pp. 714–721 (2016)
6. Jain, A., Pankanti, S.: Fingerprint classification and matching. In: Handbook for Image and Video Processing. Academic Press, London (2000)
7. United States. Federal Bureau of Investigation: The science of fingerprints: classification and uses. United States Department of Justice, Federal Bureau of Investigation (1979)
8. Sezgin, M., Sankur, B.: Survey over image thresholding techniques and quantitative performance evaluation. J. Electron. Imaging **13**(1), 146–165 (2004)
9. Parker, J.: Algorithms for Image Processing and Computer Vision. IT Pro. Wiley, New York (2010)
10. Petitcolas, F.A.P., Anderson, R.J., Kuhn, M.G.: Attacks on copyright marking systems. Lecture Notes in Computer Science, vol. 1525, pp. 15–17 (1998)
11. AlShaikh, M., Laouamer, L., Nana, L., Pascu, A.C.: Efficient and robust encryption and watermarking technique based on a new chaotic map approach. Multimedia Tools and Applications, vol. 75, pp. 1–14. Springer (2016)

Toward a Secure and Robust Medical Image Watermarking in Untrusted Environment

Abdallah Soualmi[1], Adel Alti[1], and Lamri Laouamer[2](✉)

[1] LRSD Laboratory, Department of Computer Science, University of Sétif-1,
P.O. Box 19000, Sétif, Algeria
adel.alti@univ-setif.dz

[2] Department of Management Information Systems, Qassim University,
P.O. Box 6633, Buraidah 51452, Kingdom of Saudi Arabia
laoamr@qu.edu.sa

Abstract. An untrusted environment such public Cloud as a model delivering hosted services through network. Several data are transmitted over the cloud which is considered as unsecured network, some of them may contain sensitive and important information such as medical images. In order to protect these images from several attacks, implementing a new robust digital image watermarking technique becomes necessary. In such techniques, a selected watermark is embedded in the medical image transmitted through network. The embedded watermark must be invisible and undetectable and only authorized persons can extract it to prove authentication. This paper propose a new hybrid image watermarking technique based on Number Theoretic Transform (NTT) and Diffie-Hellman on medical images in order to guarantee the security and preserve quality of medical images. We measured the robustness of the proposed approach by the commonly used metrics against several scenarios of attacks such as adding noise, rotation, etc. The tests are performed on several types of medical images.

Keywords: Security · Medical image watermarking · NTT · Diffie-Hellman
Medical data's protection

1 Introduction

Cloud computing technologies are well accepted by international health organizations and clinical professionals to reduce maintenance and operations cost. It enables them to consume computing resources on demand by preventing them from creating and managing internally infrastructures. Applications, services and multimedia data (texts, images, audios and videos) are accessed, delivered and used over the internet, instead of hard drive, and are paid for by cloud customer or provider.

The moving of services and data through Internet raises many security challenges specially data security and privacy protection that become major and serious concerns since data is located in different places. Data security is the process of keeping data secure and protected from not only unauthorized access but also corrupted access. The main focus of data security is to make sure that data is safe and away from any destructive

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 693–703, 2018.
https://doi.org/10.1007/978-3-319-74690-6_68

forces. Unauthorized access to such sensitive data or information can cause many problems such as corruption and violation of privacy. In order to guarantee the security of such data, the implementation of advance security techniques is required.

Despite the well development health IT infrastructures, there is a significant challenge that hinders the development of new robust and imperceptible watermarking technique. Several image watermarking algorithms have been proposed. Several benefits of cloud computing, the security still a main obstacle especially that the health information will be displayed in the third party where there is no transparency. The difficulty to keep this sensitive data confidential against malicious attacks will be potentially increased. However, all operations involved in image processing require decryption of the stored image to the plain text image. So, any algorithm in image processing application is applied after decryption process. Moreover, key storage may not be secured; confidentiality and integrity are not assured. To overcome this challenge, the homomorphic encryption is used to secure image processing over cloud computing. Homomorphic encryption is a technique in which arithmetic operations are carried out over encrypted data. We distinguish two categories of homomorphic encryption: Partially Homomorphic Encryption [13] and Fully Homomorphic Encryption [14].

Digital image watermarking is concerned with hiding information into a digital image. The problem that will be encountered is how to transmit medical images in a secure way through the cloud which is considered as an unsecured network for the transmission of such sensitive data?

Several image-watermarking algorithms have been proposed [1–15]. They can be categorized as *spatial* or *frequency* techniques. The spatial watermarking techniques are directly operated on image pixels which make the embedding/extracting operations computationally less. However, these techniques lack more robustness. The frequency techniques are based on data computed from transform coefficients like Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT). The embedding/extracting operations require an important computational complexity but these techniques are often more robust sine they favor low frequency components of the image [16].

This paper propose a new homomorphic encryption for securing medical images in the Cloud using a well-known mathematical approach which is Number Theoretic Transform (NTT) [11] combined with the well-known key exchange protocol Diffie Hellman [12]. NTT uses essentially arithmetic operations that reduces the complexity of computational and allows a fast convolution calculation. The Diffie Hellman key exchange protocol is a secure method for exchanging cryptographic keys. NTT is the technique used in both embedding and extracting processes using gray scale medical images as cover images and patient information as watermark while the protocol of Diffie Hellman is used to encrypt this watermark using its secret key.

The rest of the paper is organized as follows. Section 2 presents a new watermarking approach using hybrid NTT and Diffie Hellman on medical images. In Sect. 3, a large database of medical images is used to apply the proposed approach and the results are presented. Section 4 presents our conclusion and future works.

2 Proposed Watermarking Approach

This section presents a new watermarking approach based on NTT and a well-known security exchange protocol Diffie Hellman [2]. The approach contributes to sharing and transfer security of medical images and ensures the robustness of the embedded watermarks through an image in order to prove ownership. The NTT's and Diffie-Hellman effectiveness has been proven in the images loss less transmission and in convolution fast calculation. In this section, the proposed watermarking method is explained in details. The proposed watermarking framework consists of two processes: the watermark embedding process and the watermark extraction process. Our goal is to increase the robustness without affect the image quality.

The NTT is one of the most popular transform decomposition. It basically has the same form as Discrete Fourier Transform (DFT). Embedding the watermark data's into middle frequency coefficients give additional resistance to compression techniques without affecting the host image quality. The Number Theoretic Transform of a sequence $\mathrm{x} = \left\{x_n\right\}_{\mathrm{n}=0}^{N-1}$ composed of N elements defined in the Galois field, GF(q) of order q is a sequence $= \left\{X_k\right\}_{\mathrm{k}=0}^{N-1}$:

$$\mathrm{X_k} = \left\langle \sum_{\mathrm{n}=0}^{N} x_n \delta^{nk} \right\rangle_q \tag{1}$$

Where $k = 0, 1\ldots N - 1$ and δ represents the term generator of the order N, equal to the sequence length of the transform, of the field GF (q). The order N of the element δ, with $0 < N < q - 1$ is the value of the smallest positive integer p for which $\langle \delta^{\mathrm{p}} = 1 \rangle_q$.

In case of medical image and with respect to practical applications, the period N is the most important characteristic of a number theoretical transform. Therefore we have to search for modulus q for which at least one N^{-1} δ exists. In this case, the determination of N^{-1} is possible by Eq. (2):

$$\mathrm{N}^{-1} = q - ent(\frac{q}{N}) \tag{2}$$

A root δ is guaranteed if $pgcd(q, \delta) = 1$, i.e., it is not necessary that q is a prime number (case of $q = 77 = 7{\cdot}11$ or $q = 143 = 11{\cdot}13$). But it is necessary that $e = pgcd(q, \delta) = 1$ since:

$$\mathrm{N}^{-1} = q - ent(\frac{q}{N}) \tag{3}$$

So the prime numbers $\mathrm{q} = \mathrm{k}{\cdot}\mathrm{N} + 1$ will be investigated.

Diffie Hellman [12] is a mathematical algorithm that allows two parties to produce an identical shared secret, despite the fact that those parts might never have communicated with one another (Table 1). That shared secret can then be utilized to safely exchange a cryptographic encryption key through an unsecure communication channel. That key can be used subsequently to encrypt and identify the watermark.

Table 1. Fundamentals of Diffie-Hellman algorithm.

Function	Input	Operation	Output
Keys	g, p, a, b, ∈ ℙ	Choose a prime number *p* and a base *g*	Public key: a, b
		Exchange secret number *b, a* Compute $A = g^a \, mod \, p$ $B = g^b \, mod \, p$	Common secret key: $K = A^b \, mod \, p$ *or* $K = B^a \, mod \, p$
Encryption	m∈$\mathbb{Z}_n$	Compute c using public key with DES in ECB mode	c ϵ $\mathbb{Z}_n$
Decryption	c∈$\mathbb{Z}_n$	Compute m using secret key with DES in ECB mode	m ϵ $\mathbb{Z}_n$

Diffie-Hellman is a key exchange algorithm used to privately share a symmetric key between two parties. Once the two parties know the symmetric key, they use symmetric encryption to encrypt the data.

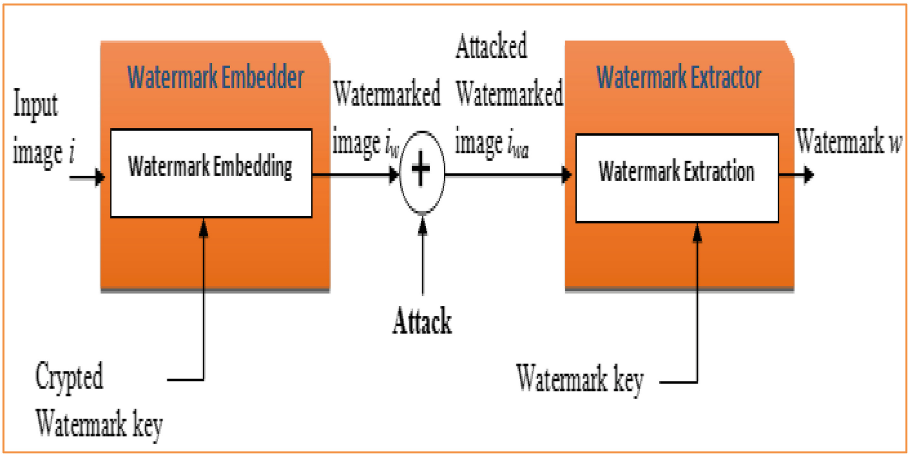


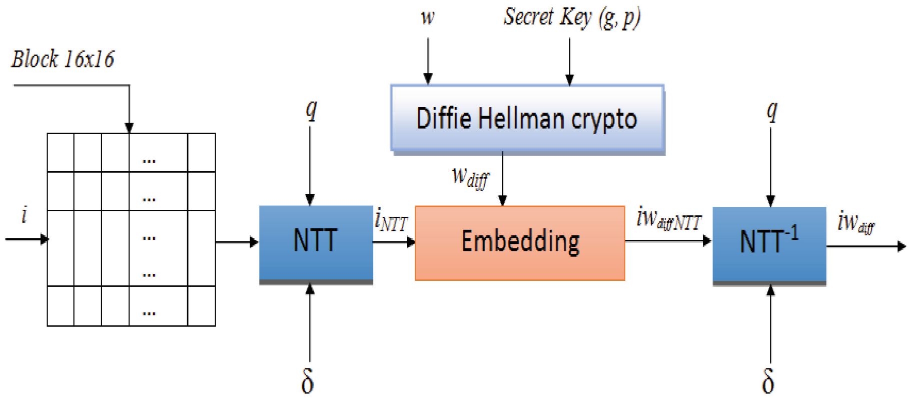
Fig. 1. Our proposed watermarking approach.

In order to guarantee the security of medical images in a public cloud that is an unsecured network, a new homomorphic encryption technique using a hybrid mathematical approach called (NTT) and Diffie Hellman is proposed. NTT is used to embed and extract the watermark. Medical images in Digital Imaging and Communications in Medicine (DICOM) format are considered as cover images. The watermark is the full name of the patient plus the name of the physician in charge. This watermark is encrypted and identified with a key that is encrypted using the protocol of Diffie Hellman. The proposed watermarking framework (Fig. 1) consists of two processes: the watermark embedding process and the watermark extraction process. Firstly, we achieve the watermark embedding based on NTT combined with the key exchange protocol Diffie Hellman to prove copyright. Secondly, the extraction process extracts the watermark after applying attacks on the watermarked image.

2.1 Watermark Embedding Process

Figure 2 illustrates the watermark embedding process. The used medical images are in grayscale of size 128 × 128. The embedding process is achieved in the frequency domain by applying the number theoretic transforms combined with the key exchange protocol Diffie-Hellman on the original image. Firstly, generating the encrypted watermark which is the full name of the patient and the name of the physician in charge based on the protocol Diffie Hellman, and then embedded it in each block of the original image. This step consists of embedding the watermark in the NTT coefficients. It is achieved in integer form carried out modulo some integer q using NTT and the generator term δ as the following steps:

1. Split the original image into 16 × 16 blocks;
2. Transform the original image by NTT transformation mode using the equation (Eq. 1).
3. The watermark is encrypted and embedded in the original image using the key encrypted with protocol Diffie Hellman algorithm.
4. NTT^{-1} is applied to get the encrypted watermarked image which is sent through unsecured network to the expert medical doctors.



where

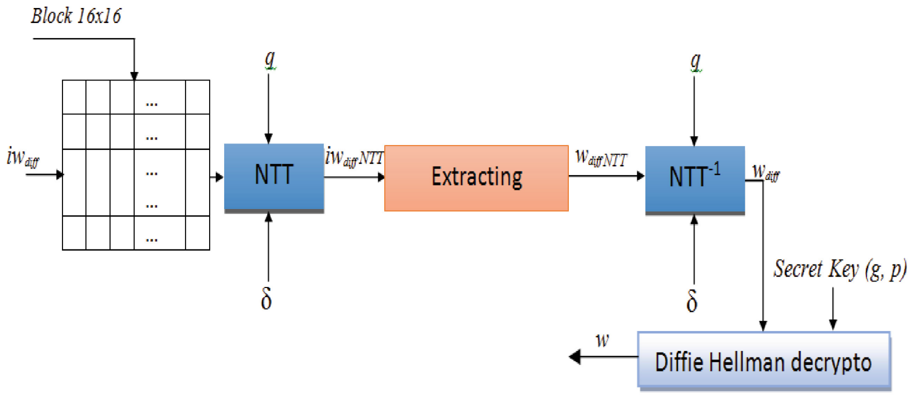
i: the original image.
w: the used watermark.
δ: the NTT generator term.
$iw_{diff\,NTT}$*: the NTT watermarked image.*
i_{NNT}*: the NTT image.*
w_{diff}*: the crypted watermark.*
q:parameter of NTT.
$iw_{diff\,NTT}$ *:the crypted watermarked image.*

Fig. 2. Watermark embedding framework.

2.2 Watermark Extraction Process

The watermarked image may be tampered in an intelligent way, in which the watermark must be imperceptible in the sense that the watermarked image has to be visually similar to the original image. The recipient, the owner, and the associated rights to the image have to be clearly identified, and an unauthorized person should not be able to generate such watermark. The watermark extraction process (Fig. 3) consists to the following steps:

5. Divide the watermarked image 16 × 16 blocks;
6. Transform the encrypted watermarked image by NTT transformation using the equation (Eq. 1).
7. The watermark is extracted from the watermarked image and NTT-1 is applied on it.
8. The watermark is image and deciphered using the key which is encrypted with the protocol of Diffie Hellman.



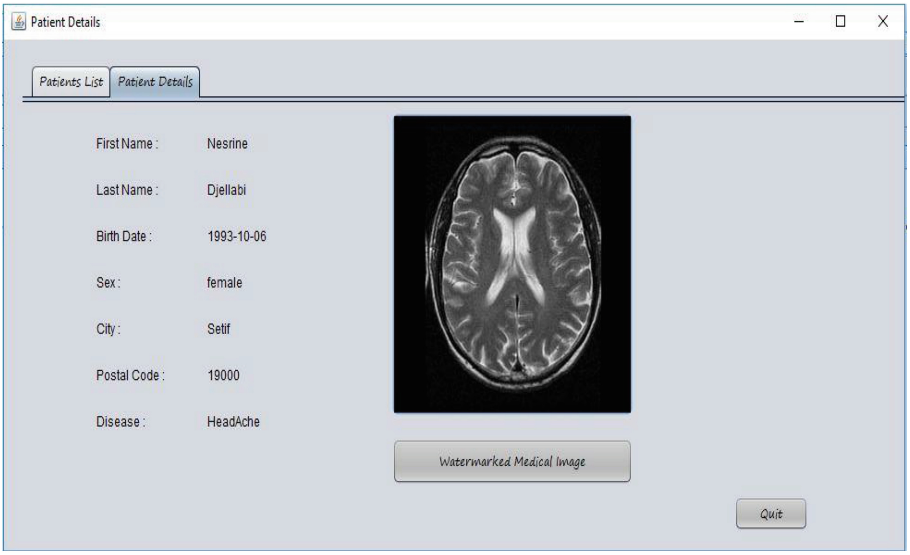
where w: the watermark
$iwdiff_{NNT}$: the NTT image

Fig. 3. Watermark embedding framework.

3 Experimental and Performance Results

3.1 Experimental Results

Our proposed approach is image watermarking with homomorphic encryption technique using NTT combined with the key exchange protocol Diffie Hellman; both are implemented using the programing language JAVA. Our prototype improves outcomes and provides helpful for authenticity and integrity of medical images. Our proposed technique is used to hide and extract the watermark using a medical image in DICOM format as a cover image and the full name of the patient plus the name of the physician in charge as the watermark. Figure 4 shows the screen shots from patient informations and Fig. 5 displays the watermarked image that will be sent from sender site. It also shows the encrypted watermark and the execution time for embedding it. Two buttons Noise and Rotation allows to apply both noise and rotation attacks on the watermarked image, it displays also the information's of the patient after extracting the watermark and the execution time needs to extract it.



First Name : Nesrine

Last Name : Djellabi

Birth Date : 1993-10-06

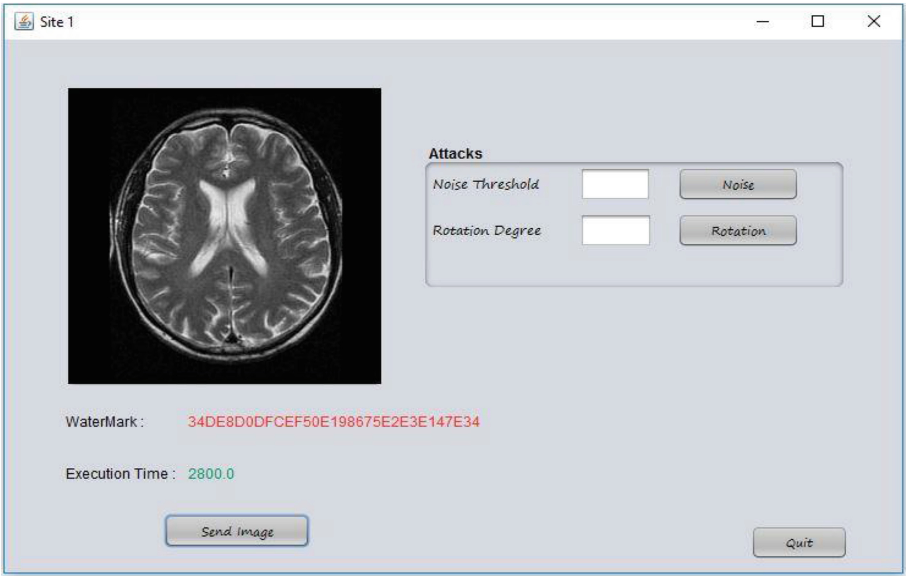
Sex : female

City : Setif

Postal Code : 19000

Disease : HeadAche

Fig. 4. Screen shots from patient informations.



WaterMark : 34DE8D0DFCEF50E198675E2E3E147E34

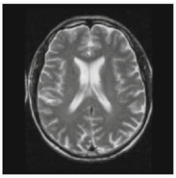
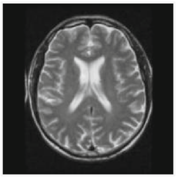
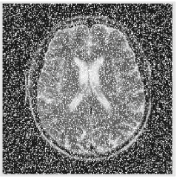
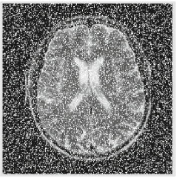
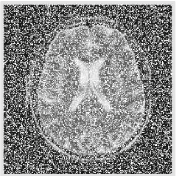
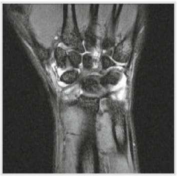
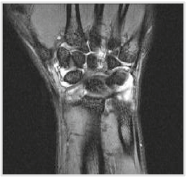
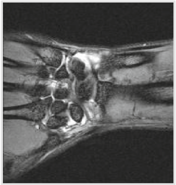
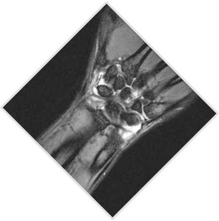
Execution Time : 2800.0

Fig. 5. Watermark embedding framework.

The tests on a gray medical image database of size 512 × 512 [17]. Our proposed watermarking algorithm is tested against two kinds of most dangerous attacks such as:

rotation and *adding noise attacks*. The watermarked images and the extracted watermarks are displayed with the execution time for embedding and extracting these watermarks in Table 2.

Table 2. Some obtained results with different noise levels and rotation degrees.

Original image	***Watermarked image***	***Attacked Image***
	 *Execution time :* ***2800ms***	 *Execution time:* ***1460 ms***  *Execution time****: 1640 ms***  *Execution time****: 1710 ms***
	 *Execution time :* ***2510ms***	 *Execution time****: 1410 ms***  *Execution time****: 1723 ms***

3.2 Performance Results

Figure 6 illustrates the execution time needed to extract the watermarks in different medical images. The time difference depends on the content of the image. The time difference depends on the content of the image. Experimental results show that our proposal approach preserves medical image quality while embedding the watermark and it shows the robustness against some attacks like noise and rotation, this is proved by extracting the original watermarks from the attacked images.

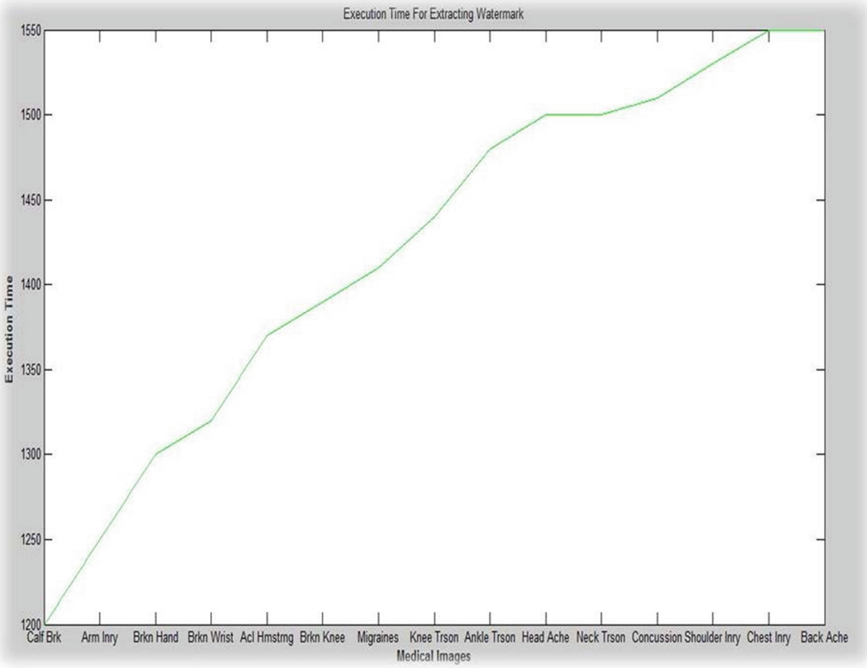


Fig. 6. Execution time measure for embedding operation.

4 Conclusion

The objective of our work was to propose an efficient watermarking technique to ensure the security of medical images transmitted through a public cloud. Our proposed approach is a hybrid of NTT technique and Diffie Hellman algorithm. This new watermarking approach allows the embedding of the watermark in a gray scale medical image using NTT technique, the embedded watermark is encrypted with the secret key of Diffie Hellman. The proposed approach with both processes of embedding and extracting that were well defined and illustrated.

The implementation of our proposed approach is realized with the programming language Java and MATLAB. The robustness of our approach against some attacks was proved with different experimental results. In future works, we will intend to test our approach robustness against standard attacks in a large database of medical images in the cloud.

References

1. Kishore, P.V.V., Kishore, S.R.C., Kumar, E.K., Kumar, K.V.V., Aparna, P.: Medical Image Watermarking with DWT-BAT algorithm. pp. 270–275. Department of E.C.E, Kula Lampur University (2015)
2. Laouamer, L., AlShaikh, M., Nana, L., Pascu, A.C.: Robust watermarking scheme and tamper detection based on threshold versus intensity. J. Innov. Digit. Images **2**(1–2), 01–12 (2015)
3. Arun, I., Kabi, K.K., Saha, B.J.: Blind digital watermarking algorithm based on DCT domain and fractal images. In: IEEE Conference on IT in Business, Industry and Government (CSIBIG), pp. 104–110 (2014)
4. Das, C., Panigrahi, S., Sharma, V.K., Mahapatra, K.K.: A novel blind robust image watermarking in DCT domain using inter-block coefficient correlation. Int. J. Electron. Commun. 01–10 (2013)
5. Ghosal, S.K., Mandal, J.K.: Binomial transform based fragile watermarking for image authentication. J. Inf. Secur. Appl. 01–10 (2013)
6. Singh, A.K., Dave, M., Mohan, A.: Multilevel encrypted text watermarking on medical images using spread-spectrum in DWT domain. Wirel. Pers. Commun. **83**, 2133–2150 (2015). Springer Science and Business Media
7. Hsu, L.Y., Hu, H.T.: Blind image watermarking via exploitation of inter-block prediction and visibility threshold in DCT domain. J. Vis. Commun. Image Represent. 01–20 (2015)
8. Su, Q., Niu, Y., Wang, Q., Sheng, G.: A blind color image watermarking based on DC component in the spatial domain. Int. J. Light Electron Opt. **124**(23), 6255–6260 (2013)
9. Vaishnavia, D., Subashini, T.S.: Robust and invisible image watermarking in RGB color space using SVD. Procedia Comput. Sci. **46**, 1770–1777 (2015)
10. Daraee, F., Mozaffari, S.: Watermarking in binary document images using fractal codes. Pattern Recognit. Lett. **35**, 120–129 (2014)
11. Laouamer, L.: Toward a robust and fully reversible image watermarking framework based on number theoretic transform. Signal Imaging Syst. Eng. Indersci. **10**(4), 169–177 (2017)
12. Kallam, S.: Diffie-Hellman: Key Exchange and public Key CryptoSystems. pp. 5–6. Master degree of Science, Math and Computer Science, Department of India State University, USA (2015)
13. Gomathikrishnan, M., Tyagi, A.: HORNS-A homomorphic encryption scheme for cloud computing using residue number system. IEEE Trans. Parallel Distrib. Syst. **23**(6), 995–1003 (2011)
14. Mohanta, B.K., Gountia, D.: Fully homomorphic encryption equating to cloud security: an approach. IOSR J. Comput. Eng. (IOSRJCE) **9**, 46–50 (2013)
15. Ansari, I.A., Pant, M., Ahn, C.W.: Robust and false positive free watermarking in IWT domain using SVD and ABC. Eng. Appl. Artif. Intell. **49**, 114–125 (2016)
16. Guikema, S.D., Aven, T.: Assessing risk from intelligent attacks: a perspective on approaches. Reliab. Eng. Syst. Saf. **95**(5), 478–483 (2010)
17. http://www.barre.nom.fr/medical/samples/

Gray Matter Volume Abnormalities in the Reward System in First-Episode Patients with Major Depressive Disorder

Qianrui Qi[1], Wei Wang[2], Zhaobin Deng[3], Wencai Weng[3], Shigang Feng[1], Dongqing Li[1(✉)], Zhi Wu[1(✉)], and Hongbo Liu[1(✉)]

[1] School of Information, Dalian Maritime University, Dalian 116026, China
{dmuldq,dmuwz}@sina.com, lhb@dlmu.edu.cn
[2] Physical Science and Technical College, Dalian University, Dalian 116622, China
[3] Affiliated Xinhua Hospital, Dalian University, Dalian 116622, China

Abstract. In current time, the crowd of depression has increased rapidly across colleges and universities in China. They often suffer anhedonia, social failure, drug abuse and so on. The recent reports points out that the brain reward system showed damaged in patients with depression, so the identification of the dysfunction in the brain reward system is really crucial. By analyzing brain magnetic resonance imaging (MRI) data, this article aimed to identify the gray matter volume (GMV) abnormalities in brain reward system between first-episode medication-naive patients with major depressive disorder (MDD) and healthy controls (HCs). 14 medication-naive participants with MDD aged 19–24 years (6 males, 8 females) and 18 healthy controls aged 19–24 years (9 males, 9 females) were recruited. We used voxel-based morphometry (VBM) to analyze brain imaging data. Then, two sample t-test was applied to detect GM abnormalities in MDD compared to HCs. This study found that MDD showed increased gray matter volume (GMV) in putamen, precuneus and amygdala in right hemisphere compared to HCs. Furthermore, MDD showed decreased GMV in left rectus, right orbital medial prefrontal cortex (omPFC), left superior temporal gyrus (STG) and left insula compared to HCs. However, no significant changes were found in caudate nucleus. These experimental results show that the depression disorder causes extensive damage in reward system and subcortical brain regions, and the alterations in the rectus, the omPFC, STG, precuneus and amygdala, may be characters of MDD in first episode.

Keywords: Major depressive disorder · Reward system
Gray matter volume · Magnetic resonance imaging
Voxel-based morphometry

1 Introduction

MDD is a psychological disorder that dose harm to human health. It has become the most common mental disease in the world with the higher and higher

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 704–714, 2018.
https://doi.org/10.1007/978-3-319-74690-6_69

morbidity [33]. According to a recently published research [31], the number of depressive patients accounts for about a quarter of all diseases in the world and will be the second most prevalent disorder by 2030. People suffering from depressive disorders experience substantial loss of quality of life, including anhedonia [1], social failure, drug abuse, and more severely suicide [9]. Especially, anhedonia is a major symptom of depression characterized by losing the abilities and motivations to experience the joy of happiness or enjoy themselves from usual activities. It is estimated that approximately 60% of all suicides have a history of depression [20]. Not only did depression cause severe damage to the patient's quality of life, but also it brought huge economic pressure on family and health care industry at the same time.

With the development and innovation of the neuroimaging technologies, such as MRI, the researches of MDD have been entering a new level, and the researches on the reward loop were explored more and more deeply [4]. The reward loop, also known as the limbic system dopamine reward circuit which considered to be associated with depression, is a neural network consisting of the nucleus accumbens, caudate nucleus, putamen, thalamus, hypothalamus, amygdala and other deep brain nuclei and medial prefrontal cortex. A paper published recently generalized the findings about the reward loop [8], implying that the cortical-basal ganglia circuit is at the heart of the reward system. It emphasized that the key structures in this circuit are the anterior cingulate cortex (ACC), the orbital frontal cortex (OFC), the ventral striatum, the ventral pallidum, and the midbrain dopamine neurons. Furthermore, some structure, involving the amygdala, hippocampus and specific brainstem regions also play significant role in rewarding [23]. In addition, most studies have demonstrated functional lesions in the reward network in depressed patient using functional MRI in last few years. Particularly, there were hyporesponsivity in striatal reward regions, frontal cortex and cingulate gyrus in MDD [28].

In this paper we choose structural MRI to explore the relationship between the dysfunction of the reward system and GMV abnormalities. Although the roles of these key regions are found, the exact pattern of the GM abnormalities in the operation of reward in depression is not defined. For instance, the striatum, as the core component of reward circuits, showed inconsistent phenomenon. Macmaster suggested that no differences were found in the caudate in adolescent major depressive disorder and bipolar depression [16]. On the other hand, depressed adolescents had smaller gray matter volume relatively to HCs in the frontal lobe and caudate nucleus bilaterally in Shad's report [26]. Although most studies have found reduction of GMV in OFC and ACC and have consistency in the function of these regions [2,7], Stratmann revealed discrepant findings that no differences existed between controls and MDD individuals [29]. Certain studies showed reduction of GMV in the amygdala [5], however others didn't find any changes in amygdala or larger volume in this area [7]. In addition, the inconsistency of hippocampus structural abnormality was also showed in previous studies. Chen and his colleagues reported decreased GMV in hippocampus [5]. Whereas, there were researches failed to detect any micro GMV differences between controls and patients in this area [6]. This kind of heterogeneity is widespread in the pathophysiology of MDD. The reasons that contribute to inconsistent and discrepant results are complicated. First, participants at various age levels are implied cross studies and the limited small number of subjects account for the variability of structural brain volume changes. Second, the usage of analytic methods may have impact on the observations of structural change in brain regions. Furthermore, illness duration and number of depressive episodes cause an intrinsic heterogeneity of studies regarding for the brain structural abnormalities in depressed patients.

On the basis of the discoveries explored by previous brain structural literature, we aimed to investigate the relationship between GMV alterations and damage in reward network in a cohort of young and well-characterized adults who are college students aged between 19 and 24 by using whole-brain analysis methods. Considering that medication has an effect on the GMV in depressed patients, this paper employed medication-naive participants undergoing MRI scanning to obtain a better result. Furthermore, because reward activities and emotion are regulated by circuits including most brain regions, such as PFC, especially in OFC, ACC, striatum areas and limbic system [8,22], we hypothesize that medication-free patients of first-episode MDD may have GMV alterations especially in OFC, ACC, striatum and amygdala areas.

2 Materials and Methods

2.1 Participants

14 medication-naive participants with MDD aged 19–24 years (6 males, 8 females) and 18 healthy controls aged 19–24 years (9 males, 9 females) were recruited. All participants were recruited from the School of Information at Dalian Maritime University from 2013 to 2016. Both patients and controls were paid a small honorarium for study participation. This study was approved by the local Ethics Committee and all participants provided written informed consent.

2.2 Magnetic Resonance Imaging Acquisition

The MRI studies were conducted with a 3.0 Tesla Siemens Trio MRI scanner in Affiliated Xinhua Hospital of Dalian University. Acquisition was performed using an 8-channel head coil. Three-dimensional T1-weighted anatomical images were acquired using a 3D SPGR sequence ($TR = 1680\,\text{ms}; TE = 285\,\text{ms}; FlipAngle = 9; TI = 500\,\text{ms}; NEX = 1; ASSET = 1.5$; Frequency direction: S/I). A total of 160 contiguous 1 mm slices were acquired with a 384 × 384 matrix, with an in-plane resolution of 1 mm × 1 mm resulting in isotropic voxels.

2.3 Statistical Analysis

SPM8 (The FIL Methods group, UK) and VBM8 (Structural Brain Mapping Group, Germany) software programs were used in both preprocessing and processing for scanned images based on MATLAB R2011a. Brain volume normalizing, bias correcting and segmentation into gray matter, white matter, and cerebrospinal fluid were performed using VBM8 toolbox. VBM8 toolbox is based on an optimized voxel-based morphometric protocol that helps to increase the signal to noise ratio. After preprocessing, SPM8 was used to smooth the images. In the whole-brain analyses, we performed a two sample t-test in the SPM8 to determine whether there were any significant regional GMV differences between MDD and healthy controls. A threshold of $p < 0.05$ uncorrected for the whole-brain volume at a cluster level was used to select statistically significant brain regions whose $clustersize > 10$.

3 Results

3.1 Group Differences in Brain GM Volumes

The VBM results showed that the gray matter density in the left rectus, the right omPFC, left STG, left insula, left ACC, left putamen nucleus and left hippocampus was significantly lower in the first episode of severe depression than in the normal control group ($p < 0.05$). The density of gray matter in the right precuneus, right amygdala and left thalamus was significantly higher than that in the normal control group ($p < 0.05$, Table 1, Figs. 1, 2 and 3).

Table 1. Brain areas with significant differences in gray density between patients with MDD and HCs

Brain regions	MNI coordinate			Cluster size	T value
	X	Y	Z		
MDD < HCs					
Rectus_L (aal)	0	63.0	−15.0	227	5.20
Frontal_Med_Orb_R(aal)	1.5	63.0	−13.5	243	4.94
Temporal_Sup_L (aal)	−61.5	−18.0	4.5	417	4.20
Insula_L (aal)	−30.0	−25.5	18.0	78	3.27
Cingulum_Ant_L (aal)	−9.0	27.0	−10.5	174	2.75
Putamen_L (aal)	−30.0	−15.0	1.5	52	2.66
Hippocampus_L (aal)	−21.0	−21.0	−12.0	54	2.23
MDD > HCs					
Precuneus_R (aal)	10.5	−43.5	43.5	162	4.40
Amygdala_R (aal)	27.0	4.5	−15	142	3.43
Thalamus_L (aal)	−1.5	−15.0	16.5	151	2.99

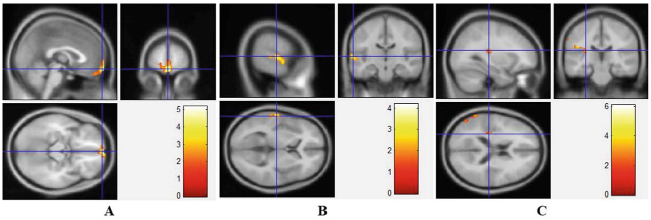


Fig. 1. Regions of gray matter volume decreases among patients with MDD as compared with HCs. (A) Gray matter volume reduced in the region of left rectus and right orbital medial frontal gyrus (peak MNI coordinate: Rectus_L: $0, 63, -15; t = 5.2$, Frontal_Med_Orb_R: $1.5, 63, -13.5; t = 4.94, p < 0.05$). (B) Gray matter volume reduced in the region of left superior temporal gyrus (peak MNI coordinate: $-61.5, -18, 4.5; t = 4.20, p < 0.05$). (C) Gray matter volume reduced in the left insula (peak MNI coordinate: $-21, -21, -12; t = 2.23, p < 0.05$). The color density represents the Tscore.

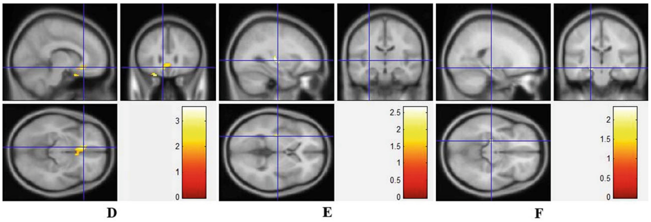


Fig. 2. Regions of gray matter volume decreases mildly among patients with MDD as compared with HCs. (D) Gray matter volume reduced in the region of left anterior cingulum (peak MNI coordinate: $-9, 27, -10.5; t = 2.75, p < 0.05$). (E) Gray matter volume reduced in the region of left putamen nucleus (peak MNI coordinate: $-30, -15, 1.5; t = 2.66, p < 0.05$). (F) Gray matter volume reduced in the left hippocampus (peak MNI coordinate: $-30, -25.5, 18; t = 3.27, p < 0.05$). The color density represents the Tscore.

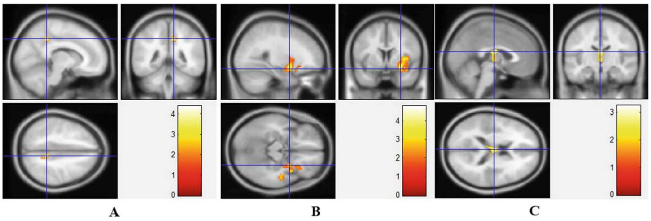


Fig. 3. Regions of gray matter volume increases among patients with MDD as compared with HCs. (A) Gray matter volume increased in the region of right precuneus (peak MNI coordinate: $10.5, -43.5, 43.5; t = 4.4., p < 0.05$). (B) Gray matter volume increased in the right amygdala (peak MNI coordinate: $27, 4.5, -15; t = 3.43, p < 0.05$). (C) Gray matter volume increased in the right thalamus (peak MNI coordinate: $-1.5, -15, 16.5; t = 2.99, p < 0.05$). The color density represents the Tscore.

4 Discussion

In the current study, we compared the GMV in reward system of depression patients and healthy subjects in a cohort of young and well-characterized adults who are medication-free college students aged between 19 and 24 by using whole-brain analysis methods, ruling out the effects of medication and aging, finding that the abnormal changes of GMV in depressive patients were bidirectional. Areas with smaller GMV in depression group than control group mainly located in the left rectus, the omPFC and the left STG, on the other hand, brain regions with significantly increased GMV are the precuneus and the amygdala in right hemisphere and the left thalamus. However, no alterations were found in caudate nucleus. The appearance of this phenomenon indicated that depressive disorder caused widespread structural abnormalities on reward circuits and fronto-limbic network [2,22], which plays key roles in the regulation of reward processing and emotion, and the significant alteration in the left rectus, right omPFC, STG, precuneus and amygdala and the slight variations in putamen nucleus and ACC may contribute to the dysfunction of rewarding and emotion in first-episode medication-naive MDD.

The left rectus and the right omPFC, which are located in the orbital frontal areas, are the most obvious areas of reduction regions and this phenomenon was consistent with our hypothesis. In a review of reward network, it emphasized that the omPFC plays a key role in the reward anticipation, such as sensory rewards and abstract rewards (e.g., money), especially near the gyrus rectus about the latter activation [8]. Moreover, the PFC, interacting with some limbic regions, also makes great contributions to the emotion processing and cognition in humans brain [25]. A meta-analysis study of depression focused on investigating brain structural changes in the cortico-striatal-pallidal-thalamic circuits, in which the PFC, especially the orbital PFC showed significantly high level of reduction of GMV. Accordingly, they approved that GMV declining in this neural circuits of depressed patients might play a significant role in the dysfunctional reward processing in depression [2]. Scheuerecker combined structural and functional researches about depressive disorders and found that decreased volume in OPFC in depressed patients by using the VBM toolkit [25]. They came to a conclusion that the OFPC is a key brain area in the emotional circuits and the alteration of the OFPC structure results in functional changes of the emotional circuits. The experimental results of the present study supported the crucial role of the OFPC in the regulation of reward, emotion and cognition and confirmed that structural changes would lead to functional changes, and furthermore hinder the expression of normal feeling.

In the current study, the putamen nucleus which is the core of the reward network and the ACC had slight GMV reduction. The putamen, critical link of the cortico-striatal-pallidal-thalamic circuits, interconnecting the cerebral cortex and subcortical areas, efficiently transmit and process emotional and rewarding information [2], and many fMRI researches stressed lower activation in response to rewarding stimulus [28]. Lai explored GMV of subcortical regions in first-episode medication-naive depressive patients and indicated that the patients had smaller volumes of the left putamen nucleus [12]. Bora et al. also suggested lower GMV in the putamen [2]. Our results about ACC were consistent with previous studies [2,32]. ACC, especially dorsal ACC, has a variety of functions, involving motivation, cognition, and motor control. Notably, the vital duty of the ACC is to monitor these functions in complex conditions [8]. Green research discovered that MDD participants showed decreased activation of the rostral cingulate gyrus during reward selection and anticipation [28], and this study supported the core role of ACC in reward system. A paper of mild depressive patients reported reduced gray matter volume in the ACC and orbitofrontal cortex using VBM analysis [32]. The reason why the extents of GMV decrease in these areas were moderate might be that the depressive participants in the current study were in the initial stage. Secondly, all subjects were in the transition period of youth to the middle-aged, eliminating the effect of aging [17]. Furthermore, education also should be taken into consideration. These discoveries indicated that the slightly decreased GMV in putamen and ACC in first-episode medication-free MDD relative to HCs might be a mark of dysfunction in reward system.

The directions of the volume changes in hippocampus and amygdala are not consistent in present study that shows decreased GMV in hippocampal and increased GMV in the amygdala. Studies of the hypothalamic-pituitary-adrenal axis (HPA) have shown that the HPA axis may have different effects on the hippocampus and the amygdala [18]. Chronic stress or trauma experiences may aggravate the burden of stress regulation and make increased activation of HPA, which lead to amounts of release of cortisol. Excessive cortisol caused two diametrically opposite effects on the amygdala and hippocampus, hypertrophy of the amygdala while atrophy of the hippocampus. This theory directly supports our results. Li studied the relationship between the severity of depression and the hippocampus and amygdala, found that the volume of amygdaloid nucleus in the mild depression group significantly enlarged and the hippocampal volume decreased in the patients with depression according to the severity of the disease (mild, moderate, major), especially in major group [15]. The participants in our experiments were college students, whose severity of depression were not so high, therefore, mild reduction of hippocampal volume and relatively more evident enlarged volume in amygdala were observed. Consequently, it might mean that overmuch stress could cause abnormalities of the amygdala and hippocampus, and gradually, their dysfunction would lead to negative emotions.

The masses of regional GMV loss in left superior temporal gyrus (STG) occurred in present study. Previous articles demonstrated the function of this region in the perception and emotion recognition and also highlighted its heavy connections with face- and body-specific cortical and subcortical structures [3]. Its interrelation with frontal-limbic circuits underpins human emotion cognition. Substantial GMV loss was present in the left STG, which is also in consistent with prior studies [13]. A meta-analysis of first-episode patients with depression revealed a significant reduction of GMV in the temporal gyrus using both the pooled meta-analysis and the subgroup meta-analysis [34]. Our findings of GMV loss in this brain region replicate previous results. With respect to this, the alterations in left STG might be a common presence in the early stage of depression.

There was also volume reduction in insula in our findings. As a key area of the limbic system, the insula has extensive connections with other brain regions about emotion processing, such as the OPFC, the ACC and the hippocampus. Ho found significantly decreased functional connectivity between insula and ACC in MDD relative to healthy controls when evaluating negative emotional stimuli [10]. Hence, the destruction of the insular function will cause tremendous damage to people's emotions, cognition, feelings and so on. Stratmann Mhad carried out experiments with more than 200 subjects and found a significant reduction of GMV in the anterior insula, and the longer the illness duration was, the more obvious the differences between groups would be [29]. Our findings that GMV reduction in insula in depressed patients were consistent with the previous articles [32]. Taken previous researches into consideration, the GMV reduction in insula may be a representative phenomenon in the course of depression. These abnormalities in the PFC, STG, insula and ACC support the notion that MDD could be conceived as a network dysfunction in brain primarily.

In this study, we also found that the GMV of precuneus was increased in depressive group compared to healthy subjects. As a core component of the default mode network (DMN), the roles of precuneus encompasses autobiographical memory retrieval, reward outcome monitoring, and emotional stimulus processing [30]. Abnormal functional connectivity in precuneus had been reported in cognitive network [27] and processing of negative stimuli [10]. Li et al. demonstrated enlarged GMV of precuneus in young women with subthreshold depression in comparison to healthy controls [14]. Another structural imaging study also reported increased gray matter density in precuneus among first-episode antipsychotic-naive MDD [35]. So that, this result may indicate hyperactivity of the precuneus maybe aggravate the symptoms of depression in MDD.

The thalamus, subcortical regions involved in the brain reward system, is a relay station for the input information in the cerebral cortex [19], actively sending message to the cortex in various mechanisms [24]. A study of single episode, medication-free MDD subjects demonstrated structural abnormalities of frontal-subcortical circuits in the early stage of MDD, such as increased GMV in left thalamus, indicating that this abnormal phenomenon may be a critical component in the mechanism of depression and excluding the influence of the medication treatment [11]. Another study about first-episode patients with MDD also concluded the unusual state in the thalamus with increased GMV [36]. A recently published article about medication-naive patients with first-episode depression also represented the similar results [21]. Accordingly, we predicted that the thalamus might play an essential role in the pathology of depression and the larger GMV in this region in MDD could be a general status.

The other areas in striatum, such as caudate, didn't show significant changes of GMV in present study. The results of previous studies on this region were not consistent. Numerous papers found reduced GMV in caudate [2,26], however no differences were detected in Macmaster research [16]. There might be several reasons. First, the durations of depression state in MDD were short in present study, and lesions of patients were not obvious. Next, analytic methods and parameters defined in this study might contribute to current result to some extent. Accordingly, more researches are needed to identify the inconsistency.

The limitations of the present study deserve mention. First, the small sample size limited the generalizability of the results. In the next stage, we should employ more participants to investigate the subtle variation between groups. Second, our cohort were college students, whose brains are undergoing GM loss during normal aging [17], affecting the comparative result to a certain extent. Furthermore, the results may not apply to older cohorts, so further researches about MDD in other age groups need to be done. Finally, other elements, such as sex and severity of depression, should be taken into consideration to precisely explore the mechanism of depression. Additionally, functional MRI, DTI and structural MRI should be study together to obtain more valuable information.

In conclusion, we demonstrated that college patients with first-episode depression, who were medication-naive, showed widespread abnormal structural GMV changes in cortical and subcortical brain areas which involved in reward circuits, especially in OPFC, amygdala, STG and precuneus. The structural abnormalities might mark medication-naive MDD in first episode. In addition to these significant variations, we also identified slight GMV alterations in other brain areas in reward system, such as putamen, ACC, thalamus and hippocampus, which may be somewhat helpful in discerning MDD of first episode.

Acknowledgment. This work is partly supported by the National Natural Science Foundation of China (Grant Nos. 61472058 and 61772102).

References

1. Admon, R., Pizzagalli, D.A.: Dysfunctional reward processing in depression. Curr. Opin. Psychol. **4**, 114–118 (2015)
2. Bora, E., Harrison, B.J., Davey, C.G., Yucel, M., Pantelis, C.: Meta-analysis of volumetric abnormalities in cortico-striatal-pallidal-thalamic circuits in major depressive disorder. Psychol. Med. **42**(4), 671–681 (2012)
3. Candidi, M., Stienen, B.M., Aglioti, S.M., De, G.B.: Virtual lesion of right posterior superior temporal sulcus modulates conscious visual perception of fearful expressions in faces and bodies. Cortex **65**, 184–194 (2015)
4. Chen, L., Zhang, W., Liu, H., Feng, S., Chen, C.L.P., Wang, H.: A space affine matching approach to fMRI time series analysis. IEEE Trans. NanoBiosci. **15**(5), 468–480 (2016). https://doi.org/10.1109/TNB.2016.2572401
5. Chen, V.C., Shen, C.Y., Liang, S.H., Li, Z.H., Tyan, Y.S., Liao, Y.T., Huang, Y.C., Lee, Y., Mcintyre, R.S., Weng, J.C.: Assessment of abnormal brain structures and networks in major depressive disorder using morphometric and connectome analyses. J. Affect. Disord. **205**, 103–111 (2016)
6. Finkelmeyer, A., Nilsson, J., He, J., Stevens, L., Maller, J.J., Moss, R.A., Small, S., Gallagher, P., Coventry, K., Ferrier, I.N.: Altered hippocampal function in major depression despite intact structure and resting perfusion. Psychol. Med. **46**(10), 2157–2168 (2016)

7. Grieve, S.M., Korgaonkar, M.S., Koslow, S.H., Evian, G., Williams, L.M.: Widespread reductions in gray matter volume in depression. Neuroimage Clin. **3**, 332–339 (2013)
8. Haber, S.N., Knutson, B.: The reward circuit: linking primate anatomy and human imaging. Neuropsychopharmacology **35**(1), 4–26 (2010)
9. Henderson, S.E., Johnson, A.R., Vallejo, A.I., Katz, L., Wong, E., Gabbay, V.: A preliminary study of white matter in adolescent depression: relationships with illness severity, anhedonia, and irritability. Front. Psychiatry **4**, 152 (2013)
10. Ho, T.C., Yang, G., Wu, J., Cassey, P., Brown, S.D., Hoang, N., Chan, M., Connolly, C.G., Henje-Blom, E., Duncan, L.G.: Functional connectivity of negative emotional processing in adolescent depression. J. Affect. Disord. **155**(3), 65–74 (2014)
11. Kong, L., Wu, F., Tang, Y., Ren, L., Kong, D., Liu, Y., Xu, K., Wang, F.: Frontal-subcortical volumetric deficits in single episode, medication-naive depressed patients and the effects of 8 weeks fluoxetine treatment: A VBM-DARTEL study. Plos One **9**(1), e79055 (2014)
12. Lai, C.H.: Hippocampal and subcortical alterations of first-episode, medication-naive major depressive disorder with panic disorder patients. J. Neuropsychiatry Clin. Neurosci. **26**(2), 142–149 (2014)
13. Lai, C.H., Wu, Y.T.: The gray matter alterations in major depressive disorder and panic disorder: putative differences in the pathogenesis. J. Affect. Disord. **186**, 1–6 (2015)
14. Li, H., Wei, D., Sun, J., Chen, Q., Zhang, Q., Jiang, Q.: Brain structural alterations associated with young women with subthreshold depression. Scientific Reports 5, 9707 (2015)
15. Li, Y., Yan, J., Wang, D., Sun, M., Zhu, Y., Zhu, X., Jiang, P., Yin, R., Zhao, L.: Magnetic resonance study of the structure and function of the hippocampus and amygdala in patients with depression. Chin. Med. J. **127**(20), 3610–3615 (2014)
16. Macmaster, F.P., Carrey, N., Langevin, L.M., Jaworska, N., Crawford, S.: Disorder-specific volumetric brain difference in adolescent major depressive disorder and bipolar depression. Brain Imag. Behav. **8**(1), 119–127 (2014)
17. Manard, M., Bahri, M.A., Salmon, E., Collette, F.: Relationship between grey matter integrity and executive abilities in aging. Brain Res. **1642**, 562–580 (2016)
18. Pagliaccio, D., Luby, J.L., Bogdan, R., Agrawal, A., Gaffrey, M.S., Belden, A.C., Botteron, K.N., Harms, M.P., Barch, D.M.: Stress-system genes and life stress predict cortisol levels and amygdala and hippocampal volumes in children. Neuropsychopharmacology **39**(5), 1245–1253 (2014)
19. Parnaudeau, S., O'Neill, P.K., Bolkan, S.S., Ward, R.D., Abbas, A.I., Roth, B.L., Balsam, P.D., Gordon, J.A., Kellendonk, C.: Inhibition of mediodorsal thalamus disrupts thalamofrontal connectivity and cognition. Neuron **77**(6), 1151–1162 (2013)
20. Peng, H., Wu, K., Li, J., Qi, H., Guo, S., Chi, M., Wu, X., Guo, Y., Yang, Y., Ning, Y.: Increased suicide attempts in young depressed patients with abnormal temporal-parietal-limbic gray matter volume. J. Affect. Disord. **165**, 69–73 (2014)
21. Peng, W., Chen, Z., Yin, L., Jia, Z., Gong, Q.: Essential brain structural alterations in major depressive disorder: a voxel-wise meta-analysis on first episode, medication-naive patients. J. Affect. Disord. **199**, 114–123 (2016)
22. Price, J.L., Drevets, W.C.: Neural circuits underlying the pathophysiology of mood disorders. Trends Cogn. Sci. **16**(1), 61–71 (2012)
23. Russo, S.J., Nestler, E.J.: The brain reward circuitry in mood disorders. Nat. Rev. Neurosci. **14**(9), 609–625 (2013)

24. Saalmann, Y.B., Kastner, S.: Cognitive and perceptual functions of the visual thalamus. Neuron **71**(2), 209–223 (2011)
25. Scheuerecker, J., Meisenzahl, E.M., Koutsouleris, N., Roesner, M., SchPf, V., Linn, J., Wiesmann, M., Bruckmann, H., MeLler, H.J., Frodl, T.: Orbitofrontal volume reductions during emotion recognition in patients with major depression. J. Psychiatry Neurosci. **35**(5), 311–320 (2010)
26. Shad, M.U., Muddasani, S., Rao, U.: Gray matter differences between healthy and depressed adolescents: a voxel-based morphometry study. J. Child Adolesc. Psychopharmacol. **22**(3), 190–197 (2012)
27. Shen, T., Li, C., Wang, B., Yang, W.M., Zhang, C., Wu, Z., Qiu, M.H., Liu, J., Xu, Y.F., Peng, D.H.: Increased cognition connectivity network in major depression disorder: a FMRI study. Psychiatry Invest. **12**(2), 227–234 (2015)
28. Smoski, M.J., Felder, J., Bizzell, J., Green, S.R., Ernst, M., Lynch, T.R., Dichter, G.S.: fMRI of alterations in reward selection, anticipation, and feedback in major depressive disorder. J. Affect. Disord. **118**(1–3), 69 (2009)
29. Stratmann, M., Konrad, C., Kugel, H., Krug, A., Schoning, S., Ohrmann, P., Uhlmann, C., Postert, C., Suslow, T., Heindel, W.: Insular and hippocampal gray matter volume reductions in patients with major depressive disorder. Plos One **9**(7), e102692 (2014)
30. Utevsky, A.V., Smith, D.V., Huettel, S.A.: Precuneus is a functional core of the default-mode network. J. Neurosci. Official J. Soc. Neurosci. **34**(3), 932–940 (2014)
31. Vavakova, M., Durackova, Z., Trebaticka, J.: Markers of oxidative stress and neuroprogression in depression disorder. Oxidative Med. Cell. Longevity **2015**(2), 898393 (2015)
32. Webb, C.A., Weber, M., Mundy, E.A., Killgore, W.D.: Reduced gray matter volume in the anterior cingulate, orbitofrontal cortex and thalamus as a function of mild depressive symptoms: a voxel-based morphometric analysis. Psychol. Med. **44**(13), 2833–2843 (2014)
33. Wjh, P.B., Yuri, M., Femke, L., Nicole, V.: Understanding the somatic consequences of depression: biological mechanisms and the role of depression symptom profile. BMC Med. **11**(1), 129 (2013)
34. Zhang, H., Li, L., Wu, M., Chen, Z., Hu, X., Chen, Y., Zhu, H., Jia, Z., Gong, Q.: Brain gray matter alterations in first episodes of depression: a meta-analysis of whole-brain studies. Neurosci. Biobehav. Rev. **60**, 43–50 (2016)
35. Zhang, J., Xiao, J., Zhu, X., Wang, X., Yao, S.: Voxel-based morphometry on grey matter concentration of the brain in first-episode, antipsychotic-naive major depressive disorder. J. Central South Univ. (Medical Science) **36**(4), 307–311 (2011)
36. Zhang, X., Yao, S., Zhu, X., Wang, X., Zhu, X., Zhong, M.: Gray matter volume abnormalities in individuals with cognitive vulnerability to depression: a voxel-based morphometry study. J. Affect. Disord. **136**(3), 443–452 (2012)

Author Index

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 715–717, 2018.
https://doi.org/10.1007/978-3-319-74690-6