Advances in Intelligent Systems and Computing 573

Radek Silhavy
Roman Senkerik
Zuzana Kominkova Oplatkova
Zdenka Prokopova
Petr Silhavy *Editors*

Artificial Intelligence Trends in Intelligent Systems

Proceedings of the 6th Computer Science On-line Conference 2017 (CSOC2017), Vol 1

Springer

Advances in Intelligent Systems and Computing

Volume 573

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

Radek Silhavy · Roman Senkerik
Zuzana Kominkova Oplatkova
Zdenka Prokopova · Petr Silhavy
Editors

Artificial Intelligence Trends in Intelligent Systems

Proceedings of the 6th Computer Science On-line Conference 2017 (CSOC2017), Vol 1

Springer

Editors

Radek Silhavy
Faculty of Applied Informatics
Tomas Bata University in Zlín
Zlin
Czech Republic

Roman Senkerik
Faculty of Applied Informatics
Tomas Bata University in Zlín
Zlin
Czech Republic

Zuzana Kominkova Oplatkova
Faculty of Applied Informatics
Tomas Bata University in Zlín
Zlin
Czech Republic

Zdenka Prokopova
Faculty of Applied Informatics
Tomas Bata University in Zlín
Zlin
Czech Republic

Petr Silhavy
Faculty of Applied Informatics
Tomas Bata University in Zlín
Zlin
Czech Republic

ISSN 2194-5357 ISSN 2194-5365 (electronic)
Advances in Intelligent Systems and Computing
ISBN 978-3-319-57260-4 ISBN 978-3-319-57261-1 (eBook)
DOI 10.1007/978-3-319-57261-1

Library of Congress Control Number: 2017937148

Printed on acid-free paper

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

Preface

This book constitutes the refereed proceedings of the Artificial Intelligence Trends in Intelligent Systems Section of the 6th Computer Science On-line Conference 2017 (CSOC 2017), held in April 2017.

Particular emphasis is laid on modern trends in artificial intelligence and its application to intelligent systems. New algorithms, methods, and applications of optimization algorithm, neural network, and deep learning and hybrid algorithms are also presented.

The volume Artificial Intelligence Trends in Intelligent Systems brings and presents new approaches and methods to real-world problems and exploratory research that describes novel approaches in the defined fields.

CSOC 2017 has received (all sections) 296 submissions, in which 148 of them were accepted for publication. More than 61% of accepted submissions were received from Europe, 34% from Asia, 3% from Africa, and 2% from America. Researches from 27 countries participated in CSOC 2017 conference.

CSOC 2017 conference intends to provide an international forum for the discussion of the latest high-quality research results in all areas related to computer science. The addressed topics are the theoretical aspects and applications of computer science, artificial intelligences, cybernetics, automation control theory, and software engineering.

Computer Science On-line Conference is held online, and modern communication technology which is broadly used improves the traditional concept of scientific conferences. It brings equal opportunity to participate to all researchers around the world.

The editors believe that readers will find the following proceedings interesting and useful for their own research work.

March 2017

Radek Silhavy
Petr Silhavy
Zdenka Prokopova
Roman Senkerik
Zuzana Kominkova Oplatkova

Organization

Program Committee

Program Committee Chairs

Zdenka Prokopova, Ph.D., Associate Professor, Tomas Bata University in Zlin, Faculty of Applied Informatics, email: prokopova@fai.utb.cz

Zuzana Kominkova Oplatkova, Ph.D., Associate Professor, Tomas Bata University in Zlin, Faculty of Applied Informatics, email: kominkovaoplatkova@fai.utb.cz

Roman Senkerik, Ph.D., Associate Professor, Tomas Bata University in Zlin, Faculty of Applied Informatics, email: senkerik@fai.utb.cz

Petr Silhavy, Ph.D., Senior Lecturer, Tomas Bata University in Zlin, Faculty of Applied Informatics, email: psilhavy@fai.utb.cz

Radek Silhavy, Ph.D., Senior Lecturer, Tomas Bata University in Zlin, Faculty of Applied Informatics, email: rsilhavy@fai.utb.cz

Roman Prokop, Ph.D., Professor, Tomas Bata University in Zlin, Faculty of Applied Informatics, email: prokop@fai.utb.cz

Prof. Viacheslav Zelentsov, Doctor of Engineering Sciences, Chief Researcher of St. Petersburg Institute for Informatics and Automation of Russian Academy of Sciences (SPIIRAS).

Program Committee Members

Boguslaw Cyganek, Ph.D., DSc, Department of Computer Science, University of Science and Technology, Krakow, Poland.

Krzysztof Okarma, Ph.D., DSc, Faculty of Electrical Engineering, West Pomeranian University of Technology, Szczecin, Poland.

Monika Bakosova, Ph.D., Associate Professor, Institute of Information Engineering, Automation and Mathematics, Slovak University of Technology, Bratislava, Slovak Republic.

Pavel Vaclavek, Ph.D., Associate Professor, Faculty of Electrical Engineering and Communication, Brno University of Technology, Brno, Czech Republic.

Miroslaw Ochodek, Ph.D., Faculty of Computing, Poznan University of Technology, Poznan, Poland.

Olga Brovkina, Ph.D., Global Change Research Centre Academy of Science of the Czech Republic, Brno, Czech Republic & Mendel University of Brno, Czech Republic.

Elarbi Badidi, Ph.D., College of Information Technology, United Arab Emirates University, Al Ain, United Arab Emirates.

Luis Alberto Morales Rosales, Head of the Master Program in Computer Science, Superior Technological Institute of Misantla, Mexico.

Mariana Lobato Baes, M.Sc., Research-Professor, Superior Technological of Libres, Mexico.

Abdessattar Chaâri, Professor, Laboratory of Sciences and Techniques of Automatic control & Computer engineering, University of Sfax, Tunisian Republic.

Gopal Sakarkar, Shri. Ramdeobaba College of Engineering and Management, Republic of India.

V.V. Krishna Maddinala, Assistant Professor, GD Rungta College of Engineering & Technology, Republic of India.

Anand N. Khobragade, Scientist, Maharashtra Remote Sensing Applications Centre, Republic of India.

Abdallah Handoura, Assistant Prof, Computer and Communication Laboratory, Telecom Bretagne, France

Technical Program Committee Members

Ivo Bukovsky
Miroslaw Ochodek
Bronislav Chramcov
Eric Afful Dazie
Michal Bliznak
Donald Davendra
Radim Farana
Zuzana Kominkova Oplatkova
Martin Kotyrba
Erik Kral
David Malanik
Michal Pluhacek
Zdenka Prokopova
Martin Sysel

Roman Senkerik
Petr Silhavy
Radek Silhavy
Jiri Vojtesek
Eva Volna
Janez Brest
Ales Zamuda
Roman Prokop
Boguslaw Cyganek
Krzysztof Okarma
Monika Bakosova
Pavel Vaclavek
Olga Brovkina
Elarbi Badidi

Organizing Committee Chair

Radek Silhavy, Ph.D., Tomas Bata University in Zlin, Faculty of Applied Informatics, email: rsilhavy@fai.utb.cz

Conference Organizer (Production)

OpenPublish.eu s.r.o.
Web: http://www.openpublish.eu
Email: csoc@openpublish.eu

Conference Website, Call for Papers

http://www.openpublish.eu

Contents

A Genetic Algorithm to Improve Lifetime of Wireless Sensor Networks by Load Balancing

Nazila Karkooki[1], Mohammad Khalily-Dermany[2](✉),
and Pouria Polouk[3]

[1] Khomein Branch, Islamic Azad University, Khomein, Iran
nazilakarkooki@yahoo.com
[2] Young Researchers and Elite Club, Khomein Branch,
Islamic Azad University, Khomein, Iran
md.khalili@gmail.com
[3] Kermanshah Branch Sama, Islamic Azad University, Kermanshah, Iran
pouria.polouk@gmail.com

Abstract. Wireless Sensor Networks (WSNs) are a collection of a large number of small sensors capable of sensing the environment. In spite of limited resources in WSNs, they are employed in various applications and a large researches are done to extend their performance. In addition to decreasing energy consumption, some strategies should be employed to balance network load and consequently balance the energy consumption of these nodes and ensure a maximum network lifetime. In this paper, with the goal of reducing energy consumption and extending lifetime, a regular network is considered, then we formalize the network lifetime as an optimization programming. By using of load balancing technique, it will increase node lifetime. However, solving this problem is complex and time consuming, so we propose a genetic algorithm. We compare optimal solution and genetic algorithm and conclude through the results that combining load balancing with energy consumption improve network lifetime.

Keywords: Wireless sensor network · Lifetime · Genetic algorithm · Load balancing

1 Introduction

One of the most important tools for getting information and understanding the environment is Wireless Sensor Network (WSNs). Such networks are made up of nodes with the capabilities of sensing, signal processing, wireless communicating ability and limited energy battery. Every sensor gathers information through sensing its area around, the collected data is transmitted to the sink through wireless transmitting [1]. In WSNs, grid topology can offer reliable communication connectivity and coverage, and different source routing protocols can be offered based on the regularity and predictability of topology [2].

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_1

Typically, short energy of each node is one of the basic problems in these networks, and limitation of energy resources causes shortening the lifetime of them. In addition, node's position sometimes will intensify the problem. For example, a one-hop node, close to the base station node, immediately loses its energy because of greater traffic load. On the other hand, the breakdown of such nodes causes breaking the connection of the nodes in the entire network and thus hampered network. All sensor nodes produce packets and send them a single repository by multi-hop transmission. In other words, where the sensor data must be sent to a base station, traffic pattern is not uniform and puts the heavy burden on sensor nodes that are near the base station [3]. An algorithm based on game theory proposed in [4] for optimizing energy consumption in WSNs. In fact, these models, energy optimizing according to the game theory, determine the distribution form of cluster-heads based on remained energy of sensors [4]. Nodes that are near to the sinks are featured with heavy transmission loads, so these performance bottleneck nodes are prone to running out of energy early with the shortened network lifetime [5].

In order to gather information more economical in terms of energy consumption, WSNs are partitioned into clusters [6]. To solve this problem, unequal clustering algorithms make clusters of different sizes, the clusters close to the base station have smaller sizes than clusters far from the base station, a fuzzy energy-aware unequal clustering algorithm was proposed in [7]. Khalily-Dermany et al. proposed a joint optimal design of topology control and network coding in [8]. They formulated the problem of topology control in network-coding-based-multicast WSN with the delay and reliability constraints as an optimization problem. Khalily-Dermany et al. make a theoretical determination of the achievable maximum multicast information flow after diverse topology control mechanisms in [9]. Their simulation results show topology control mechanisms decrease both energy consumption and maximum information flow. In [10], Elhoseny et al. proposed a genetic algorithm based method that optimizes heterogeneous sensor node clustering. Their proposed method greatly extends the network life, and the average improvement with respect to the second best performance based on the first-node-die and the last-node-die is 33.8% and 13%, respectively. The balanced energy consumption greatly improves the lifetime and allows the sensor energy to deplete evenly [11]. Bouabdallah showed that sending the traffic generated by each sensor node through multiple paths, instead of a single path, allows significant energy conservation. Specifically, they derive the set of paths to be used by each sensor node and the associated weights that maximize the network lifetime [12]. Kacimi et al. presented the lifetime maximization problem in "many-to-one" and "mostly-off" WSNs. In such network pattern. They show that when the entire sensor data has to be forwarded to a base station via multi-hop routing, the traffic pattern is highly non-uniform, putting a high burden on the sensor nodes close to the base station [13, 14]. Take into consideration that energy efficiency and lifetime have been considered in the most researches on WSNs for instance, Khodabakhsi and Khalily-Dermany in [15] proposed a solution as an algorithm for reducing energy consumption in WSNs to prolong the lifetime.

Therefore, one of the most important issues in WSNs is the problem of energy constraints and optimal energy management. This study analyzes the balancing of energy consumption and a genetic algorithm is proposed that prevent from imbalanced traffic patterns. In other words, we improve distribution of energy consumption by using a load balancing technique and genetic algorithm in the grid WSNs. In the proposed algorithm, a set of initial solutions is defined accidentally in the form of the initial population, then any member of the initial population can seek to find the globally optimal directly or with the help of other search space members.

The rest of the paper is structured as follows. Problem statement and formulation is described in Sect. 2. Section 3 demonstrates proposed genetic algorithm. Performance evaluation is described in Sect. 4 and finally in Sect. 5, we present the conclusion of this paper.

2 Problem Statement and Formulation

In this paper, the network is considered to be a grid balanced graph, a base station is placed at the corner of deployment area. All nodes on the network are placed regularly on a matrix, having the M rows and M columns and the distance between nodes is fixed, then $N = M \times M$ nodes placed, each node has its own traffic vector. This model is utilized in some recent researches such as [13].

In this model, a step-by-step routing and load sharing between the accessible nodes. Probability mode is used to send data to the neighbors and the data moves only toward the base station. Since the movement of data is just toward the base station, so any node has three single-step neighbors at most. Since the network has a grid topology, sensor nodes have two different transmission range d and $\sqrt{2}d$, so each node can employ transmission range $TPL1$ for distance d and $TPL2$ for $\sqrt{2}d$. According to the transmission level distance, it is clear that $E(TPL1) = \sqrt{2}E(TPL2)$ [14].

Traffic vector means some data is sending or receiving. Let Λ denotes a vector of outgoing traffic ratios of all nodes in the network and $\lambda_g^{(i)}$ denotes traffic generated by node i. Let P_{ji} denotes the power consumption to receive traffic sent from node j to i. The traffic is sent by the node i, includes its own public traffic and traffic received from other single-step nodes. In other words, the total traffic of a node includes the node's public traffic, traffic received from the other nodes and traffic sent to neighbor nodes according to the transmission level. Overall traffic of each node leads to consuming power in it is specified as follows:

$$E(P) = (\lambda_g^{(i)} + \sum_{j} \Lambda^{(j)} P_{ji}) * (\sum_{j} \Lambda^{(i)} P_{ij}.d) \tag{1}$$

To increase the lifetime, energy consumption of the nodes should be decreased, where each node occurs this case to send and receive minimal traffic. In this case, the node's energy consumption is decreased and consequently increasing the lifetime of the network is caused. Thus the objective function of this problem can be formulated by:

$$E_{optimal} = \min E(P) \tag{2}$$

3 Proposed Genetic Algorithm

In the previous section, we describe an optimization problem (2), however, solving this problem is time consuming and complex. So, in the following a genetic algorithm is proposed that solve it very quickly. The basic idea behind this method is derived from the Darwin's Theory of Evolution. In fact, using this method causes steps faster forward to find possible answers among the state space of the problem, and compare and rank the answers.

Note that, the network is modeled as a grid topology. By using of the proposed genetic algorithm, traffic of nodes is controlled. By using fitness function, the most appropriate strings are chosen for doing combination and mutation for transmitting to the next generation, and this action is repeated to get to the desired population. By passing time and continuing the process, nodes of the network form their routing tables and update them.

The genetic algorithm, as an optimizing computation algorithm, consider a set of solution space points in each computational repetition, effectively searches different areas of solution space. Three actions done on data include selection, combination and mutation.

3.1 The Way of Creating Chromosome and Early Population

The primary factor that is very important in genetic algorithm is the creation of the needed population for running the program. In other words, the important factor in genetic algorithm is the way of making and coding chromosomes. The suggested algorithm creates the primary population accidentally, and chromosome components are random floating point numbers between 0 and 1. Each node probably selects one node as the next one–step neighbor. Then the output traffic from each node to all single-step neighbors can be computed. The possible rate of selecting the next one–step neighbor is known as a gene that has the scale between 0 and 1. According to the fact that the network is Grid and all data is forwarded to the base station, each node has a maximum of three single-step neighbors. The length of each chromosome is considered the most $(N-1)*3$, that N is the total number of nodes within a network. Each node transmits traffic to its one–step neighbor with the probability from 0 to 1, then for each node the entrance traffic is computed from its single-step neighbor in addition to its own general traffic.

3.2 Combination Operators

The combination operator uses chromosomes of primary population and combines them together. The combination operator does the combination of chromosomes that may be the result child chromosome would be better than previous chromosomes. The chromosome that is produced after combination, should satisfy the condition, as the total possibility of transmitting by its each node will be 1. Supposedly, for a network with nine nodes that depicted in Fig. (1), one combination example is shown in the

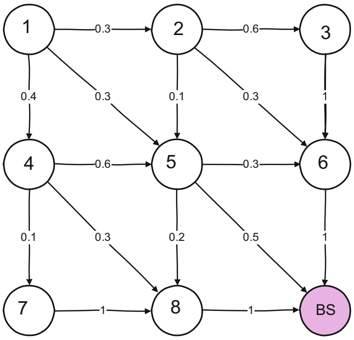
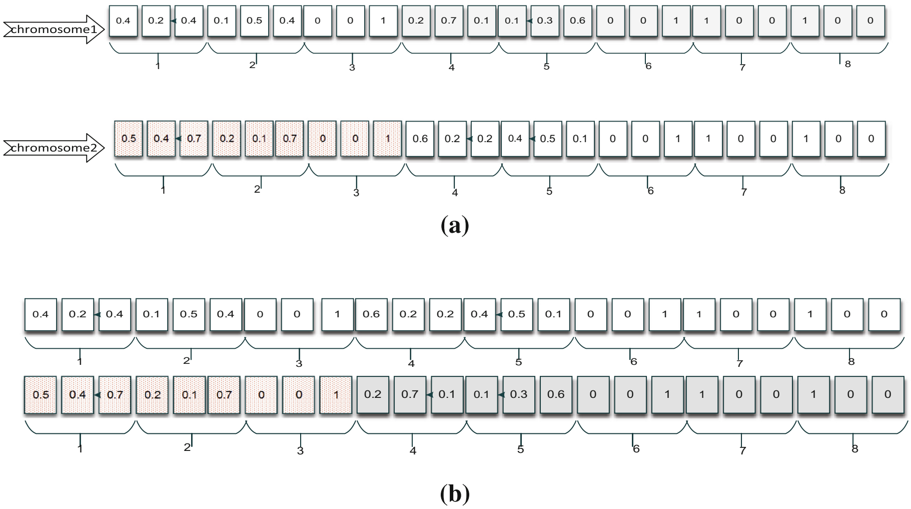


Fig. 1. Sample graph with 9 nodes



(a)

(b)

Fig. 2. (a) Choosing two chromosomes for combination (b) Forming two chromosomes after combination

Fig. 2(a). First, two chromosomes randomly will be chosen. Then they combined together from the third node. The achieved chromosomes after combination from node 3, are shown as Fig. 2(b). While two chromosomes have participated in the combination, certainly the result of the combination will be two chromosomes that are shown in the Fig. 2(b).

3.3 Mutation Operator

The mutation causes a change in intact spaces. It can be deduced that the most important task of mutation avoiding the convergence to a local optimum. Since the genetic algorithms follow the evolutionary rules, in this algorithm, the mutation

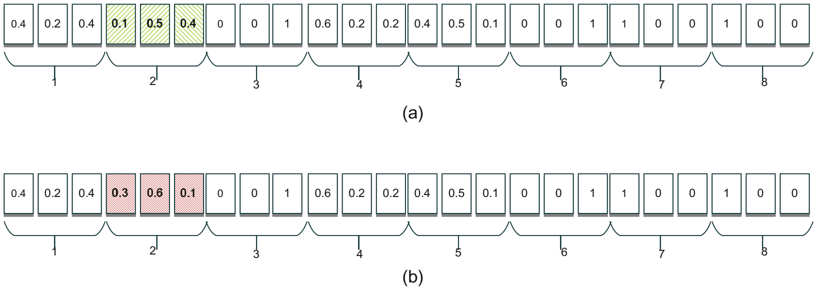


Fig. 3. (a) The participating chromosome in mutation (b) The changes of chromosome after mutation.

operator is used with fewer possibilities. In the suggested algorithm, the place of chromosomes that is considered for mutation, is chosen randomly. According to the research condition, a mutation is done if the total possibility of transmitting among nodes is 1 in chromosomes. Considering the network has 9 nodes, the participating chromosome in the possibility of mutation is shown in the Fig. 3(a). After applying mutation on the present chromosome in the Fig. 3(a), the result of a mutation is shown in the Fig. 3(b).

3.4 Fitness Function

This operator determines the amount of optimizing of each chromosome. The Suitability operator attributes a possibility to each chromosome that this probability is the same possibility of combination chromosome in future generations. Clearly, more optimized chromosomes will have more chances for combining with the other chromosomes.

The fitness function is the appropriate conversion of the aim function that is going to be optimized. The fitness amount of an answer will be more and the probability of participation in producing the next generation will be increased. Note that, the selecting model for this study is the roulette wheel model. The fitness function is considered equal to $E(P)$ for a chromosome.

4 Performance Evaluation

In this section, the results of the simulations are presented. All simulations are done by Matlab 2013 on a personal computer with a core i5 CPU and 4 GB of RAM. We compare proposed algorithm with optimization problem and Simulated Annealing that were proposed in [10]. In the simulation study, nodes deployed in a square region of measuring 100 × 100 area with 100 nodes. The network environment has only one sink node which is located at the corner of the region, sensor nodes are located at regular interval. Sink node and sensor nodes are fixed and we assume sink node has unlimited energy.

Table 1. Parameters of simulations

Parameter	Value
The population size	100
Crossover rate	0.8
Mutation rate	0.03
The number of iterations	200
Initial traffic	3

The population size of the genetic algorithm influences the performance of the algorithm respectively. The larger value of crossover lets us search effectively in solution space and reduce the chances of falling into the local optimum. But much larger value of it causes to waste the computation time in the space where the probability of existing solution in it is very low. On the other hand, if the mutation rate is very small, some chromosomes never generate. Meanwhile, the big mutation rate disorders the generated population and the children have a small likeness with parents. It leads to genetic algorithm never be able to learn from the search history. The probability of mutation, combination and other options of genetic algorithm, as Table 1 are considered.

Table 1 shows the initial parameters and their values in our simulations. Here, the considered topology is a two-dimensional network. We assess energy consumption when the number of nodes is added sequentially. In the considered meta-heuristic algorithm, the resulting amount is accidental, and it is possible that the results of research will be different in various simulations, so all tests in this study are repeated 20 times, and the average of results are reported in Fig. 4. Figure 4 displays the effect of the number of chromosomes when the number of nodes equal to100. Clearly, when the number of iterations increase, the proposed algorithm gradually converges to the optimal energy and after 350E4 iterations the best energy consumption obtained. In the experiment, the impact of population size on the performance of the algorithm is

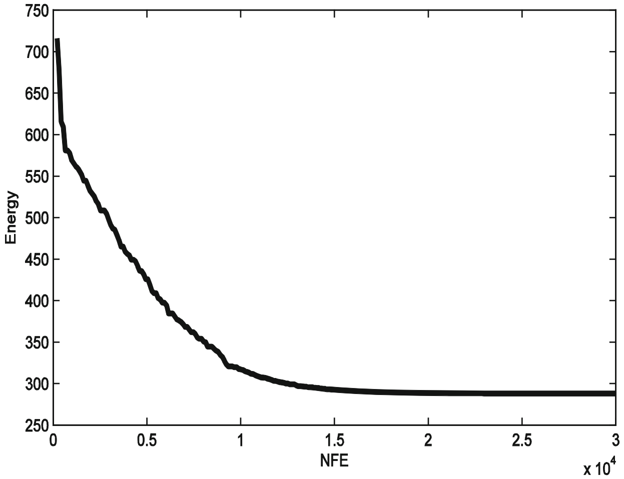


Fig. 4. The effect of the number of iterations to achieve optimal result

described. Of course, for different parameters, the amount of the cost function and energy consumption would change.

Then, we assess the effect of network size on energy consumption and the number of nodes is added sequentially and the amount of energy is computed. Results are showed in the Fig. 5. Note that the traffic (λ_g) is considered equal to 1. It is clear, by increasing the number of nodes, the amount of consumed energy raise, and our proposed algorithm is compared with optimized algorithm and Simulated Annealing. The proposed algorithm consumes less energy when compared by Simulated Annealing algorithm and is close to the optimal results.

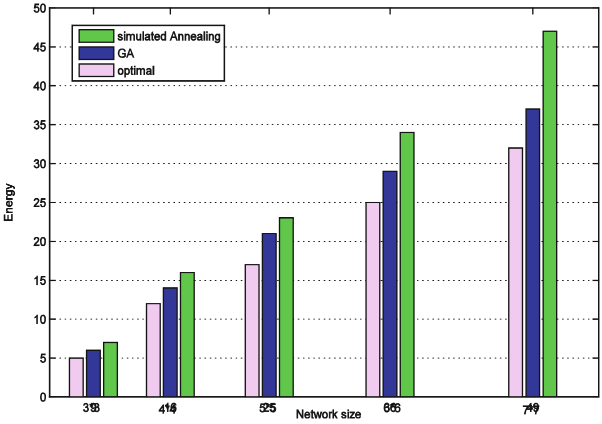


Fig. 5. The effect of change in dimension of the network

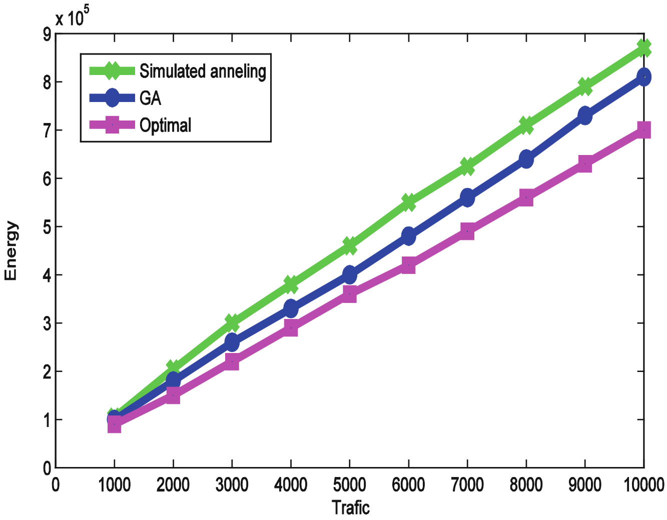


Fig. 6. The effect of increasing traffic load on the network

To better understand of the traffic load on the lifetime, more simulations are done. Figure 6 shows energy consumption according to public traffic for each node (λ_g). In this experiment, a network with 100 nodes is considered where each node generates a specified traffic load that gradually changed from 1000 to 10000. Clearly, the energy consumption increases linearly when traffic load raise. This experiment shows that obtained energy consumption is close to the optimized mode and it is better than the Simulated Annealing algorithm.

5 Conclusion

The energy consumption of sensors is very important and unbalanced energy consumption will reduce the lifetime of WSNs. In this paper, we propose an optimization problem that determines routes such that maintain node's energy and balance traffic data and subsequently increase lifetime. The complexity of solving optimization problem is high, so, a genetic algorithm is proposed. The proposed algorithm is a problem-solving method using nature and the principles governing it. By utilizing proposed approach, traffic will be balanced, energy consumption will be decreased and cause to increase lifetime. In the proposed genetic algorithm, best neighbor nodes are selected in terms of traffic and energy consumption. We assess proposed algorithm through simulation and simulation results show proposed algorithm improve energy efficiency. However, simplicity, speed and cost efficiency are the main advantages of proposed approach.

References

1. Azharuddin, M., Jana, P.K.: Particle swarm optimization for maximizing lifetime of wireless sensor networks. Comput. Electr. Eng. **51**, 26–42 (2016)
2. Khalily-Dermany, M., Shamsi, M., Nadjafi-Arani, M.J.: A convex optimization model for topology control in network-coding-based-wireless-sensor networks. Ad Hoc Netw. **59**, 1–11 (2017)
3. Zhao, F., Guibas, L.J.: Wireless Sensor Networks: An Information Processing Approach. Morgan Kaufmann, San Francisco (2004)
4. Zheng, G., Liu, S., Qi, X.: Clustering routing algorithm of wireless sensor networks based on Bayesian game. J. Syst. Eng. Electron. **23**(1), 154–159 (2012)
5. Li, W., et al.: Performance comparison of source routing tactics for WSN of grid topology. In: 2014 IEEE 12th International Conference on Dependable, Autonomic and Secure Computing (DASC). IEEE (2014)
6. Hamzeloei, F., Khalily-Dermany, M.: A TOPSIS based cluster head selection for wireless sensor network. Procedia Comput. Sci. **98**, 8–15 (2016)
7. Bagci, H., Yazici, A.: An energy aware fuzzy approach to unequal clustering in wireless sensor networks. Appl. Soft Comput. **13**(4), 1741–1749 (2013)
8. Khalily-Dermany, M., Sabaei, M., Shamsi, M.: Topology control in network–coding–based–multicast wireless sensor networks. Int. J. Sens. Netw. **17**(2), 93–104 (2015)
9. Khalily-Dermany, M., Sharifian, S.: Effect of various topology control mechanisms on maximum information flow in wireless sensor networks. SmartCR **5**(1), 10–18 (2015)

10. Elhoseny, M.: Balancing energy consumption in heterogeneous wireless sensor networks using genetic algorithm. IEEE Commun. Lett. **19**(12), 2194–2197 (2015)
11. Darehshoorzadeh, A., Javan, N.T., Dehghan M., Khalily-Dermany, M.: LBAODV: a new load balancing multipath routing algorithm for mobile ad hoc networks. In: 6th National Conference on Telecommunication Technologies and 2008 2nd Malaysia Conference on Photonics, pp. 344–349 (2008)
12. Bouabdallah, F., Bouabdallah, N., Boutaba, R.: On balancing energy consumption in wireless sensor networks. IEEE Trans. Veh. Technol. **58**(6), 2909–2924 (2009)
13. Kacimi, R., Dhaou, R., Beylot, A.-L.: Load-balancing strategies for lifetime maximizing in wireless sensor networks. In: 2010 IEEE International Conference on Communications (ICC). IEEE (2010)
14. Kacimi, R., Dhaou, R., Beylot, A.-L.: Load balancing techniques for lifetime maximizing in wireless sensor networks. Ad Hoc Netw. **11**(8), 2172–2186 (2013)
15. Khodabakhshi, B., Khalily-Dermany, M.: An energy efficient network coding model for wireless sensor networks. Procedia Comput. Sci. **98**, 157–162 (2016)

Proposal of a Fuzzy Control System for Heat Transfer Using PLC

Martin Nesticky(✉), Tomas Skulavik, and Jaroslav Znamenak

Faculty of Materials Science and Technology in Trnava,
Slovak University of Technology in Bratislava,
Paulinska 16, 917 24 Trnava, Slovak Republic
{martin.nesticky,tomas.skulavik,jaroslav.znamenak}@stuba.sk

Abstract. The article presents a fuzzy controller design for using PLC as a control system. The subject of controlling is a mathematical model of a heat exchanger modeled in Simulink. The control algorithm was developed and configurated in the FuzzyControl++ configuration tools for the automation of technical processes. The goal of the paper is to explore the possibilities to implement and realize the potential of fuzzy control in areas belonging to the domain of classical regulation. Model of the closed loop control system in virtual environment is used to perform multiple simulations to detect the system behavior under various conditions. It is concluded that the applied control strategies can offer same properties in progress of regulation and offer better approach to intuitive knowledge base design and configuration parameters of controller.

Keywords: Fuzzy control · Programmable logic controller · Heat exchange

1 Introduction

Efficient control is tightly related to improvements in the quality of industrial production processes. In complex plants, a choice has to be made between the various available strategies (conventional, fuzzy and neural) developed in last decades. In the industry, fuzzy logic is implemented in non-linear closed-loop controllers, response prognostics of mathematically complex processes or a process automation application that cannot be solved with existing standard tools. Empirical process know-how and verbalized knowledge by experience can be directly transformed into controllers, patter identification or decision logic. Numerous intuitive software programs, which require only minimal specialist knowledge about fuzzy logic, allows to use fuzzy logic in many fields and applications of industrial automation.

The model of an indirect heat exchanger and a Fuzzy control system is designed in this paper. For the designing process of the exchanger model, Matlab Simulink was used. Simulink is widely used to spread engineering ideas, including developing safety-critical software [1], designing industrial communication

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_2

systems [2], designing controllers for various systems [3,4], and others. Fuzzy Logic Toolbox software provides the direct implementation of the control system into a simulation as in [5]. Nowadays, many processes operated by humans are automated using various control techniques. Conventional control technique performance, such as PID control, in itself is often inferior to that of the human operator. One of the reasons is that linear controllers, which are commonly used in conventional control, are not appropriate for nonlinear plants. Another reason is that humans aggregate various kinds of information and combine control strategies, that cannot be integrated into a single analytic control law.

Fuzzy logic can capture the continuous nature of human decision processes and as such is a definite improvement over methods based on binary logic (widely used in industrial controllers). Programmable Logic Controller systems were used typical for discrete (event) control – automotive, electronics, etc. Their primary goal was to replace the relay technology. Nowadays, wide instruction libraries including function block for continues control (well-designed PID, lead-lag blocks, etc.) together with fuzzy toolboxes for PLCs and a universal fuzzy system for PLC with a possibility to convert Matlab fuzzy system into PLC's fuzzy structure do exist [6].

2 Materials and Methods

In this paper, mathematical model of controlled system represent heat transfer in indirect heat exchanger. The model is based on template system presented in [7]. The heat exchanger consist of two tanks. The tanks shares one wall as a partition. This partition is used for heat transfer and has surface area S. Warmer liquid A is in the first tank, with input temperature T_1 and output temperature T_{10}. Colder liquid B is in the second tank, with input temperature T_2 and output temperature T_{20}. In the system is a flow valve, which regulates the flow of the liquid A into the first tank (Fig. 1).

In the considered system, q_1 denotes the amount of the fluid A in-flowing to the first tank with volume of V_1. The fluid B in-flows to the second tank with the amount of q_2. The second tank has volume V_2. The changes of temperatures of fluids in the tanks are expressed by the equation

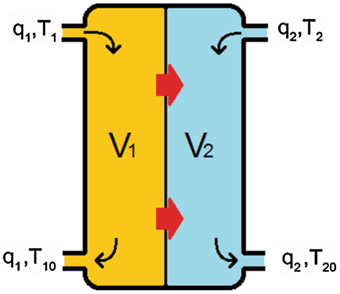


Fig. 1. Indirect heat exchanger

$$\frac{dT_{10}}{dt} = \frac{q_1}{V_1}T_1 - \frac{q_{10}}{V_1}T_{10} - \frac{k_s S}{\rho_1 c_{p1} V_1}(T_{10} - T_{20}), \tag{1}$$

$$\frac{dT_{20}}{dt} = \frac{q_2}{V_2}T_2 - \frac{q_{20}}{V_2}T_{20} + \frac{k_s S}{\rho_2 c_{p2} V_2}(T_{10} - T_{20}), \tag{2}$$

where k_s is thermal conductivity of partition, ρ is density of liquids and c_p is specific heat capacity of liquid. From these formulas, following block diagram was composed (Fig. 2). To build the model in Matlab Simulink it was necessary to take into account the adjustable input amount of the fluid q_1 using the input flow valve (Fig. 3). Controlling of the temperature of the fluid B in the second tank is made by adjusting q_1, the inflow of fluid A to the first tank. Blocks Divide modules were added to the block diagram, which enable setting input flow q_1 using the fuzzy controller. The required information about characteristics of the system was obtained from the modified model. The information was used to correct the design of the fuzzy controller.

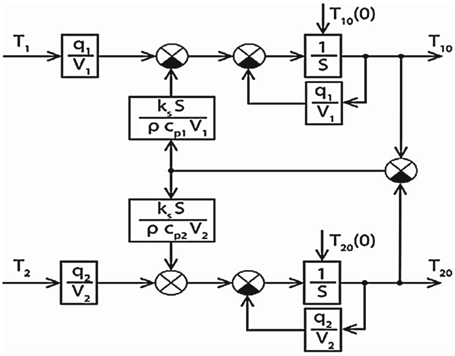


Fig. 2. Block diagram of indirect heat exchanger

All parts of closed loop control system can be designed in Simulink, but it is possible to send some data out of the Simulink model for calculations. Figure 4 shows the model of controller. The error signal e and the change of the error de are obtained, their values are converted and then sent to the PLC. The PLC uses fuzzy logic to calculate the control signal u, which is send back to the Simulink. To imitate system with flow valve as a regulator, Limited Integrator was added as a subsystem (Fig. 5) where the function $f(u)$ is as follows:

$$f(u):\ u[2] \times (((u[1] > 0) + (u[2] >= 0)) > 0) \times (((u[1] < 1) + (u[2] <= 0)) > 0) \tag{3}$$

Proposal and testing the fuzzy control of the heat exchanger was carried out in Matlab Simulink. PLC systems is now used to implement the fuzzy logic in the control process. It is possible to implement fuzzy logic and fuzzy control

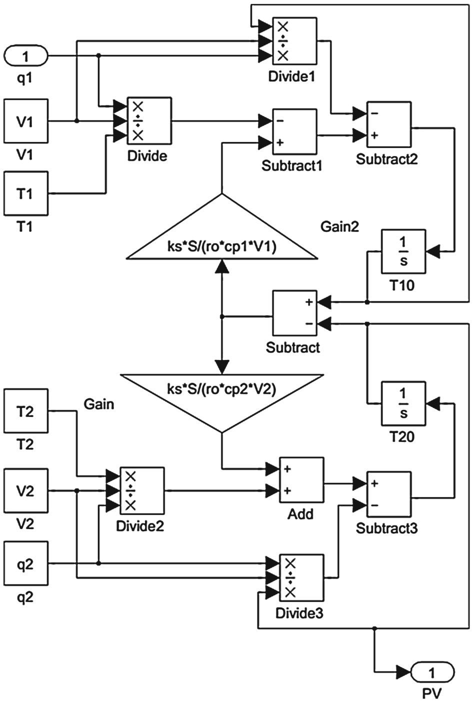


Fig. 3. Matlab model of indirect heat exchanger

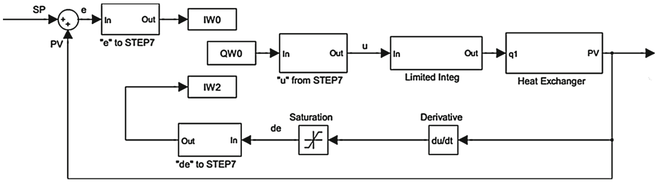


Fig. 4. Feedback control loop

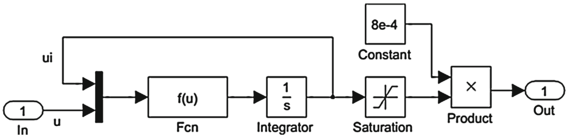


Fig. 5. Limited integrator

in industrial controllers with FuzzyControl++, a Siemens configuration tool. Development and configuration of fuzzy system is divided into two steps. In first step, the model of control system is modified in FuzzyControl++. In second step, created data are downloaded to the PLC and the parameters are adjusted more precisely. Developing and testing of the PLC fuzzy controller was carried on in the virtual SIEMENS PLC S7-300. The processor was CPU315F-2 and MPI interface was used to communicate with FuzzyControl++. Real-time connection between Matlab Simulink and the PLC program was provided by the PLCSIM blockset. In the PLC, the function block FB30 contains all the fuzzy algorithms and procedures. The program automatically creates the data block DB30. This data block contains all the variables and the parameters which represent the structure of the fuzzy controller. The function block contains also the memory elements, which are required to call function block FB30. It was necessary to take into account, that if the Simulink model of heat exchanger uses data type of INTEGER, the input and output from function block FB30 is only data type of REAL. Therefore, conversion of input and output values are required in PLC program. The fuzzy controller was proposed as a Sugeno type and was designed in the FuzzyControl++.

This controller has two inputs (error signal e and derivation of the error signal de), and it has one output signal (the control signal u). Both of the input signals e and de are defined by three membership functions. More functions does not improve the accuracy of control but proportionally increase the computation time. The functions were drafted in Membership Function editor using a Gaussian curve for each input signal. The error signal e was in the interval from −20° C to 20° C (Fig. 6) and the change of the error de was in the interval from −0.1° C to 0.1° C (Fig. 7). The control signal u is output from designed fuzzy module and this output signal is characterized by five membership functions *close_full, close_more, without_change, open_more, open_full.* The function *close_full* fully closes the flow control valve and the function *open_full* fully opens the valve. The functions *close_more* and *open_more* control the valve setting in small steps. The function *whitout_change* makes no change on valve setting. Features of controller also follows from fuzzy rules. Proper reaction of controller is dependent on the appropriate design of fuzzy rules. The rules are created and

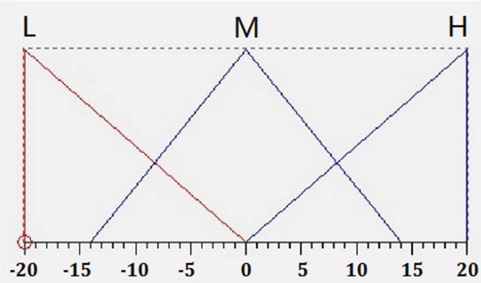


Fig. 6. Input variable “e”

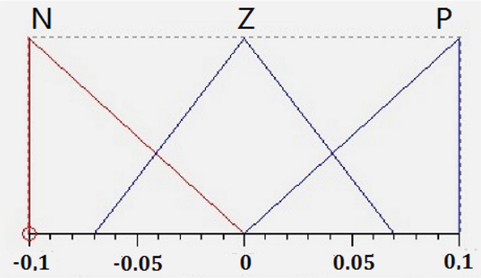


Fig. 7. Input variable “de”

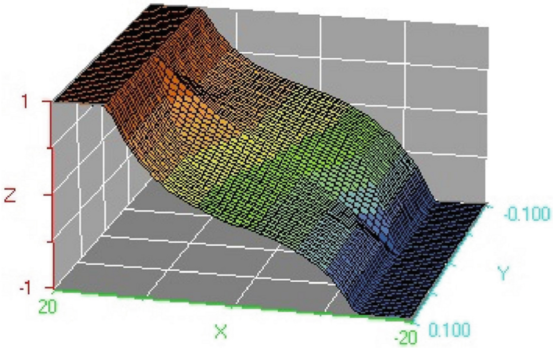


Fig. 8. 3D representation of fuzzy rules

edited in the editor of fuzzy rules. The fuzzy rules for calculating the value of the control signal u are as follows (Fig. 8):

- If (e is M) and (de is Z) then (u is *whitout_change*)
- If (e is M) and (de is P) then (u is *close_more*)
- If (e is M) and (de is N) then (u is *open_more*)
- If (e is L) then (u is *close_full*)
- If (e is H) then (u is *open_full*)

3 Results and Discussion

Sugeno fuzzy controller has been tested in several stages. At the beginning, proposed design of controller has to be tested. During this test, the control process was not affected by any error from outside and set point was static in time. The result of this test was correct response of the control signal to difference between set point and actual value of the output temperature of the liquid B. If the desired and the actual temperature were equal, then the control signal stays in its stabilized value. If the difference between the desired and the actual temperature were positive or negative then the control signal was decreasing or increasing. If the actual temperature was in the appropriate proximity of the set point then dynamical characterization was taken into consideration. Whether change of the actual temperature was positive or negative, the control signal was increasing or decreasing.

The next stage involved testing in closed loop controlling of heat transfer. Initial condition of model parameter was set according the Table 1. Dynamical characterization of the actual temperature of the fluid B and desired temperature as a set point were compared. If controller failed in expected demands, then parameters of membership function were adjusted. Several simulations with constant values of a set point were performed and the progress of process variable was monitored.

Table 1. Input parameters of the system

Symbolic name	Initial value	Description
ρ	$1000\,\mathrm{kg/m^3}$	Density of liquids
k_s	$1000\,\mathrm{Wm^2/K}$	Thermal conductivity of partition
V_1	$0.03\,\mathrm{m^3}$	Volume of the 1st tank
V_1	$0.04\,\mathrm{m^3}$	Volume of the 2nd tank
S	$2\,\mathrm{m^2}$	Surface area of partition
c_{p1}	4195 kJ/(kg K)	Specific heat capacity of liquid A
c_{p2}	4183 kJ/(kg K)	Specific heat capacity of liquid B
q_1	$0.0001\,\mathrm{m^3/s}$	Rate of flow of liquid A
q_2	$0.00021\,\mathrm{m^3/s}$	Rate of flow of liquid B
T_1	80° C	Input temperature of liquid A
T_{10}	80° C	Output temperature of liquid A
T_2	20° C	Input temperature of liquid B
T_{20}	20° C	Output temperature of liquid B

After fulfilling all requirements, the system was tested using a variable value for set point instead of constant value. In the model this signal was generated by the Signal Builder block. The signal was changing in steps and reaction of the fuzzy controller was monitored. Ability of system reaction in time shows Fig. 10. The test started with initial set temperature of 30° C.

At the beginning, the desired temperature was 30° C. Overshoot appeared but it did not indicate the wrong functionality of the controller in this case. The transfer heat between liquids takes some time, so the controller responses to temperature changes gradually. Because of the inertia of the controlled system, it incurs a delay between the actual temperature and desired temperature. The fuzzy controller was also testing on ability to react on disturbance from outside. In the time 9000 s the rate of flow of liquid B decreased as show Fig. 9. Figure 10 shows the increase of real temperature due to the disturbance. The controller

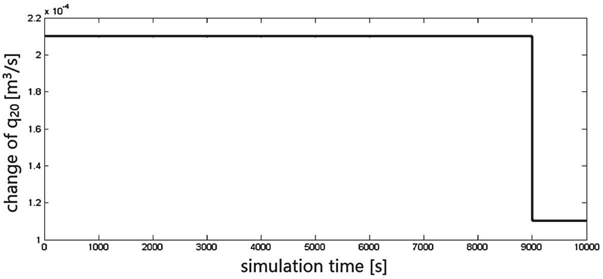


Fig. 9. Trend of real temperature (PV) and desired temperature (SP) with disturbance at 600 s

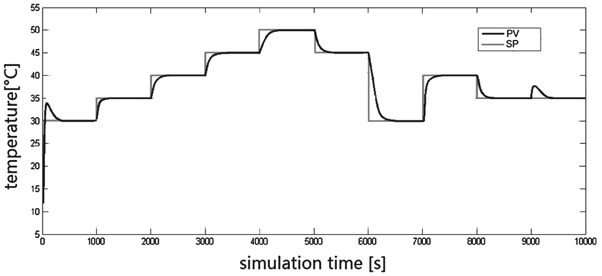


Fig. 10. Trend of actual temperature (PV) and desired temperature (SP)

responded correctly and in the time 9200 the real temperature and desired temperature were equal.

4 Conclusion

This study applied a fuzzy system for controlling heat transfer in indirect heat exchanger. The fuzzy system determines operating control parameters of heat exchanger based on the current states. The approach with the expert knowledge could obtain efficient and near-optimal solutions when compared to the simulation-optimization approach. The Fuzzy Control System was implemented in Programmable Logic Controller.

The virtual plant models and simulation models with other principles support companies in identifying and implementing Industry 4.0 scenarios. Based on the engineering experience, the fuzzy logic technologies have been developed to consider the problems of optimization and decision making in the presence of uncertainty. Many applications of fuzzy logic in industry have been successfully developed.

Acknowledgments. The authors would like to thank for financial contribution from the STU Grant scheme for Support of Young Researchers.

The research work is supported by the Project KEGA 040STU-4/2016 Modernization of the Automatic Control Hardware course by applying the concept Industry 4.0 (Aplikacia konceptu Industry 4.0 v ramci modernizacie predmetu Technicke prostriedky automatizovaneho riadenia).

This publication is the result of implementation of the project: "UNIVERSITY SCIENTIFIC PARK: CAMPUS MTF STU - CAMBO" (ITMS: 26220220179) supported by the Research & Development Operational Program funded by the EFRR.

References

1. Rastocny, K., Zdansky, J.: Specificities of safety PLC based implementation of the safety functions. In: Proceedings of International Conference Applied Electronics, Pilsen, 5–7 September 2012, pp. 229–232 (2012). ISBN 978-80-261-0038-6, ISSN 1803-7232, IEEE Catalog Number: CFP1269A-PRT
2. Franekova, M., Kallay, F., Peniak, P., Vestenicky, P.: Communication security of industrial networks. Monography, EDIS ZU, Zilina (2007). ISBN 978-80-8070-715-6
3. Tothova, M., Pitel, M.: Reference model for hybrid adaptive control of pneumatic muscle actuator. In: 9th IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI 2014), pp. 105–109. IEEE Press, Timisoara (2014)
4. Tothova, M., Pitel, M., Mizakova, J.: Electro-pneumatic robot actuator with artificial muscles and state feedback. Appl. Mech. Mater. **460**, 23–31 (2014)
5. Tóthová, M., Pitel, J., Hošovský, A.: Simulation of hybrid fuzzy adaptive control of pneumatic muscle actuator. In: Silhavy, R., Senkerik, R., Oplatkova, Z.K., Prokopova, Z., Silhavy, P. (eds.) Intelligent Systems in Cybernetics and Automation Theory. AISC, vol. 348, pp. 239–246. Springer, Cham (2015). doi:10.1007/978-3-319-18503-3_24
6. Korosi, L., Turcsek, D.: Fuzzy system for PLC. Slovak University of Technology, Faculty of Electrical Engineering and Information Technology, Institute of Control and Industrial Informatics (2011)
7. Noskievic, P.: Modelovani a idendifikacia systemu, pp. 116–118. Montanex, Ostrava (1999). ISBN 80-7225-030-2

The Study of Improved Particle Filtering Target Tracking Algorithm Based on Multi-features Fusion

Hongxia Chu[1,2(✉)], Zhongyu Xie[2], Du Juan[2], Rongyi Zhang[2], and Fanming Liu[1]

[1] College of Automation, Harbin Engineering University,
Harbin, Heilongjiang, China
chx0420@163.com

[2] College of Electrical and Information Engineering,
Heilongjiang Institute of Technology, Harbin, Heilongjiang, China

Abstract. In view of the shortcomings of traditional particle filter which is lacking of utilizing current observational information, this paper proposes a multi-featured fusion tracking algorithm based on simulated annealing to improve particle filter. The proposed method solves the problem of large amount of computation and lack of particle number in high dimensional state. A hierarchical random search annealing method is used to generate a better proposal distribution in the Monte Carlo importance sampling. In the likelihood approximation, this paper integrated image feature attribute of colors and edges to generate weight function in the different annealing layer by weighting. Using this method to track the moving objects with complex background and occlusion, the experimental results show that the proposed method has high tracking accuracy and strong stability.

Keywords: Particle filtering · Proposal distribution · Simulated annealing · Multi-features fusion

1 Introduction

Research of target tracking in video sequence is an important and challenging task within the field of computer vision field. The core thought is to detect, track and distinguish targets as well as to describe and analyze them with computer vision technique in image sequences. The research is widely used in intelligent video surveillance, robot vision, human-computer interfaces and safety examination areas. The difficulty of target tracking centralizes on picture-noise influence, illumination variation, clutter, unstable target features, occlusion, posture variation and so on. So it is a challenge to design a high-speed robust target tracking algorithm. The representative target algorithm is particle filtering [1]. The key procedure is the confirmation of proposal distribution. The closer proposal distribution is from the posterior probability distribution, the better properties of the particle filtering. Traditional particle filter utilizes priori probability density function as the proposal distribution. It has the advantages of easy calculating weight and convenient sampling of the proposal probability density. Because the

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_3

proposal probability density has no relations with present quantity-measurements, efficiency of the particle is low. It can't solve the problem of calculating a large amount of particles and particle number degeneration under high-dimensional conditions. Some scholars combined particle filter with mean-shift to restrain the particle degeneration in some extent. A large number of particles are required in particle tracking, so it is difficult to assure real-time [2]. Some scholars combined multi-features within the frame of particle filter to improve accuracy and stability of the tracking [3–5]. Also, some scholars integrated color information with motion direction and other information within the frame of particle filtering, and it gets well experiment tracking effects [6]. Simulated annealing algorithm is a multi-mode random optimization method based on a probabilistic search approach and is seldom used in tracking documents. Jonathan Deutscher [7–9] used the information integration thought and applied annealing into particle filter. Then the performance of particle filter has been improved significantly for tracking the human body. Simulated annealing (SA) is a probabilistic method proposed by Metropolis for finding the global minimum of a cost function that may possess several local minima. SA, as an extension of partial search algorithm, produced a new state model randomly in the process of amending models. Nature mechanism introduction not only lets simulated annealing receive target function "better" test point in iteration procedure, but also lets simulated annealing receive target function "poor" test point according to certain probability. States in iteration process are random, and not demand the later states should be better than the former ones. Therefore, SA is easy to intervene to the existing model. It is extensibility and easy to combine with other technology. The idea of SA algorithm is introduced to PF. One-time state of the PF algorithm was transferred for changing process of particle state under the control of the temperature. Overall energy state of the PF system is an equilibrium with mutual restraint and mutual of the thermal motion effect inside the particle.

On the basis of the above analysis, in the light of defect of traditional particle filtering proposal distribution, which lacks the utilizing of current observation information, a kind of improved multi-feature integration annealing proposal distribution methods is proposed within the frame of particle filter video tracking application. Weighting function is produced by applying the image feature properties of the fusion between colors and edges to weight in different annealing layers. The combination bond is the calculation of particle weighting value. The comparison of experimental effects between traditional particle filter and improved annealing particle filter tracking was provided.

2 Particle Filtering

Particle filter draws out N individual distribution samples $\left\{x_{0:k}^{(i)}\right\}$ by utilizing the Monte Carlo method from the posterior probability density function $P(x_{0:k}|z_{1:k})$ of state. Posterior density function (PDF) of state can be approached as by empirical distribution

$$\hat{p}(x_{0:k}|z_{1:k}) = \frac{1}{N}\sum_{i=1}^{N}\delta(x_{0:k} - x_{0:k}^{(i)}) \tag{1}$$

However, PDF is unknown in general. At this time, N samples $\left\{x_{0:k}^{(i)}\right\}$ are needed to be individually drawn out from an important distribution function $q(x_{0:k}|z_{1:k-1})$ which can easily be sampled. PDF can be similar formulated as:

$$\begin{cases} \hat{p}(x_{0:k}|z_{1:k}) = \sum\limits_{i=1}^{N} \tilde{\omega}_k^{(i)} \cdot \delta(x_{0:k} - x_{0:k}^{(i)}) \\ \tilde{\omega}_k^{(i)} = \omega_k^{(i)} / \sum\limits_{i=1}^{N} \omega_k^{(i)} \end{cases} \tag{2}$$

Where, $\omega_k^i = \omega_{k-1}^i \frac{p(z_k|x_k^i)p(x_k^i|x_{k-1}^i)}{q(x_k^i|x_{k-1}^i, z_{1:k})}$ can be regarded as important weight value. System state estimation on K time is

$$\widehat{x}_k = \sum_{i=1}^{N} \widetilde{\omega}_k^{(i)} x_k^{(i)} \tag{3}$$

$q(x_k^i|x_{k-1}^i, z_{1:k})$ is proposal distribution (important density) function. Selecting proposal distribution is very important in the whole process. The most simple and easy to implement approach is to make it equal to the prior density, that is $q(x_k^i|x_{k-1}^i, z_{1:k}) = p(x_k^i|x_{k-1}^i)$. At this time, $\omega_k^i = \omega_{k-1}^i p(z_k|x_k^i)$. It's obvious that the method hasn't considered the latest observation value. There is a comparatively big deviation between the samples drawn from the important function and the one generated by true posterior distribution. When the distribution of the likelihood function is narrow or there are a few overlaps between the distribution of prior density and the measurement likelihood function, only a small number of particles can get bigger weight values. So it makes more particles abandoned in the re-sampling procedure and aggravates the particle degeneration. It maybe leads to the failure of particle filtering. Aim to this defect, the proposal distribution function is selected through simulation annealing thought. After improvements, annealing particle filtering will not depend on the model. Even though model lacks of precision or observation noise becomes louder, this reference distribution can also effectively express the real distribution. Meanwhile, the linearism of systematic state equation is unnecessary when updating samples. Particle filter can really accomplish non-linearity and solve the problem of particle degradation.

2.1 System State Description and State Transferring Model

The purpose of video track is to get position coordinate and dimensional information of motive target. The rectangle can be used to describe an interesting area. For motive targets, it is difficult for a random model to satisfy motive description, so introducing speed weight. At the same time, the components of the width, height, and weight of the target are introduced in order to meet target change. Then target state vector can be expressed by one six-dimensional vector. It can be parameterized as $x = \{x, y, \dot{x}, \dot{y}, s_x, s_y\}$.

Where, x and y is centroid coordinates of the rectangle. $\dot{x}$ and $\dot{y}$ is the velocity of targets along x and y. s_x *and* s_y is the width and height of targets. We use first-order auto-regressive (AR) equation to define dynamic models. It can be formulated as

$$x_k = Ax_{k-1} + v_k \tag{4}$$

Where, A is a systematic state transferring matrix. v_k is process noise.

3 Annealing Particle Filtering

3.1 The Weighting Function

A number of factors must be taken into account when deciding which image features are to be used to construct the weighting function. Firstly, the used image features should be invariant under a wide range of conditions so that the same tracking framework will function well in a broad variety of situations. Secondly, in an effort to make the tracker as efficient as possible the used features must be easy to extract. Three image features were chosen to construct the weighting function: color and edge features and texture.

Color is an important feature of the target. This article uses the “kernel” concept referred in mean shift algorithm to create a color histogram [10]. The weighted color histogram is used as a color distribution model of target. In RGB space, color histograms are calculated with many small color bits. In our experiments, 8 × 8 × 8 bits are sufficient to represent the color distribution for pixels with 8-bit color depth in each channel. When constructing color distribution models for a half-length and half-width rectangular area, weighting function is selected according to the different contributions pixel in different areas towards color histograms.

$$k(r) = \begin{cases} 1 - r^2 & r < 1 \\ 0 & r \geq 1 \end{cases} \tag{5}$$

In this, r is the distance from some points to the regional center. Thus, particle color histograms which regard y as the center candidate region can be expressed as:

$$p_u(y) = C_h \sum_{i=1}^{N} k(\left\| \frac{y - x_i}{h} \right\|^2)\delta[b(x_i) - u] \tag{6}$$

Where, C_h expresses normalization constant. N is the total pixel number of the target region. u is the index value of histogram segments. $b(x_i)$ means the instruction function of pixel point x_i in its histogram. $\delta(\cdot)$ is the Kronecker delta function. $h = \sqrt{h_x^2 + h_y^2}$ describes the size of the target area.

Similarity is measured by the Bhattacharyya distance between color histogram $p_u(y)$ in candidate model and color histogram q_u in the target model. That is

$$d_c = \sqrt{1 - \rho(p_u, q_u)} \tag{7}$$

$$\rho(p_u, q_u) = \sum_{u=1}^{M} \sqrt{p_u(y), q_u} \tag{8}$$

Equation 7 is the discrete Bhattacharyya coefficient. Color histogram is more similarity between candidate model and target model when d_c becomes smaller gradually. Similarity likelihood observation function distance for distance is modeled as a Gaussian distribution [11, 12].

$$w^{co} = w_{color}(Z_{color,k}|x_k) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{d_c^2}{2\sigma^2}) \tag{9}$$

3.1.1 Edge Feature

In complicated situations, single color feature information doesn't contain any motion and shape information. Also, it's easily influenced by illumination variation and clutter. Edge feature, as another important feature, can effectively adapt to illumination variation. Thus, we regard edge feature as the second tracking feature and add edge feature information into an object model. That means, oval or rectangular with parameters is regarded as shape models. Suppose in the experiment, we use rectangle contour to select target area, thus we need count similar function of shape cue according to the rectangle. The similar accountant of the rectangle is using one point p, and then drawing one measuring line from this p point towards the center of the rectangle. Along this line, there are n Fixed-interval sampling points around the center point p. On each point, similar function is count with the canny edge detector. Suppose real edge point distributions are standard, that the mean is zero and the variance is σ^2. Then the similar function of observation sampling points are:

$$w_{shape}(z_{shape,k}^{(l)}|x_k) = 1 + \frac{1}{\sqrt{2\pi}\sigma h_0} \sum_{j=1}^{n_l} \exp(-\frac{(z_j - x)^2}{2\sigma^2}) \tag{10}$$

h_0 is priory probability of being measured unreal edge. Z_j is the distance from being measured feature point to the rectangle. Then all the similar functions m lines with even distribution around rectangle can be formulated as [13]

$$w^e = w_{shape}(z_{shape,k}|x_k) = \prod_{l=1}^{m} P_{shape}(z_{shape,k}^{(l)}|x_k) \tag{11}$$

3.1.2 Texture Feature

Texture feature is an important tracking feature of the target description. It reflects the properties of the image itself and has a strong mutability of the anti-light photo. But also texture features based on the gray level co-occurrence matrix have the ability of anti

1	2	3
4	5	6
7	8	9

Fig. 1. Pixel distribution map

noise and occlusions. Its extraction is simple and processing speed is fast. It meets the requirements of the target tracking in real time and accuracy. It can make up that the color feature is easily affected by illumination and occlusion by edge feature. The description of texture features based on the gray level co-occurrence matrix is shown in Fig. 1.

It assumes that the pixel value at position five is $G_{x,y}$. The transverse direction is x, and the longitudinal direction is y. Gray difference was calculated between each pixel and the pixel in the direction of 45°, 135°, 90° and 0° for the selected tracking target area.

$$\begin{aligned} G_1(x,y) &= G(x+1,y+1) - G(x-1,y-1) \\ G_2(x,y) &= G(x-1,y+1) - G(x+1,y-1) \\ G_3(x,y) &= G(x,y+1) - G(x,y-1) \\ G_4(x,y) &= G(x+1,y) - G(x-1,y) \end{aligned} \tag{12}$$

The extracted gray co-occurrence matrix is calculated through two order statistics, and mean matrix is used to compute G_5:

$$G_5(x,y) = [G_1(x,y) + G_2(x,y) + G_3(x,y) + G_4(x,y)]/4 \tag{13}$$

Similar to color histogram, gray histogram of image based on texture feature is obtained. Then texture similarity d_t is obtained between the target template and the candidate target. So the similar observation likelihood function of texture feature is:

$$w^t = w_{texture}(Z_{texture,k}|x_k) = \frac{1}{\sqrt{2\pi}\sigma}\exp(-\frac{d_t^2}{2\sigma^2}) \tag{14}$$

3.1.3 Multiple Cues Integration of Annealing Particle Filtering

Particle filter provides a good probability framework for integration tracking. Any probability observation models can accomplish tracking tasks within the framework. In order to satisfy robust tracking requirements, we use former color, edge and texture cues. Weighting function is generated by weighting the image features property of color and edge integrated in different annealing layers, so that the proposal distribution is improved. The counting integration link is the account of particle weighting values.

According to Reference [14, 15], we suppose the observation statistic of each cue is individual, then the integrated similar weighting function on the x_k state can be formulated as

$$w^{\beta(l)} = \exp\left(-\sum_{c=1}^{C} (\lambda_c^{co}(\beta(l), l)w^{co} + \lambda_c^{e}(\beta(l), l)w^{e} + \lambda_c^{t}(\beta(l), l)w^{t} \right) \tag{15}$$

In Eq. (15) $\beta(l)$ is the annealing rate. The weight coefficient λ_c is calculated according to the annealing rate and characteristic importance. w^{co}, w^{e}, w^{t} is the observation function based on color, edge and texture. We call feature fusion based on the annealing. Its structure is shown in Fig. 2.

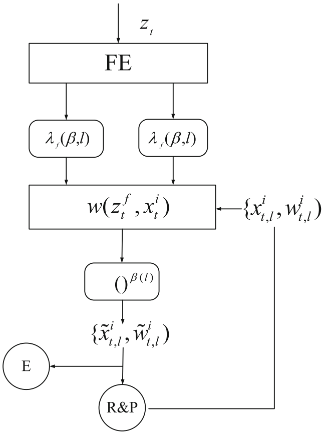


Fig. 2. Feature-based annealing PF

3.2 Annealing Procedure

Presented particle tracking method based on simulated annealing is different from traditional particle tracking method, this method generates partial particle using the posterior distribution of target color features through systematic state transferring equation. Meanwhile, it generates a weighting function with edge and texture particle features and applies image feature attribute of colors and edges to generate weight function at different annealing layer by weighing.

A series of weighting functions $w_0(Z, X)$ to $w_M(Z, X)$ are employed in which each w_m differs only slightly from each other. The function w_m is designed to be very wide, representing the overall trend of the search space while w_0 should be very peaked, emphasizing local features. Expression way can be formulated as:

$$w_m(Z, X) = w(Z, X)^{\beta_m} \tag{16}$$

In this formula, $\beta_0 > \beta_1 > \cdots > \beta_M$, however, $w(Z, X)$ is the original weighting function.

One annealing procedure is achieved through image observation value z_k in each time t_k. The state of the tracker after each layer of an annealing procedure is represented by a set of N weighted particles $S^{\pi}_{k,m} = \{(s^{(0)}_{k,m}, \pi^{(0)}_{k,m})\ldots(s^{(N)}_{k,m}, \pi^{(N)}_{k,m})\}$. But an unweighted set of particles can be described as $S_{k,m} = \{(s^{(0)}_{k,m})\ldots(s^{(N)}_{k,m})\}$. In $S^{\pi}_{k,m}$, each particle is regarded as a $(s^{(i)}_{k,m}, \pi^{(i)}_{k,m})$ pair. $\pi^{(i)}_{k,m}$ is the corresponding particle weight. Each annealing procedure can be described as follows:

(1) About each time step $t_{k,}$, annealing procedure begins on M layer, and $m = M$.
(2) Each layer in annealing run is initialized by a set of un-weighted particles $S_{k,m}$.
(3) Then, each of these particles is allocated with a weight

$$\pi^{(i)}_{k,m} \propto w_m(Z_k, s^{(i)}_{k,m}) \tag{17}$$

(4) N particles are drawn randomly from $S^{\pi}_{k,m}$ with replacement and with a probability equal to their weight $\pi^{(i)}_{k,m}$. Particle $s^{(n)}_{k,m-1}$ is generated by choosing the n^{th} particle $s^{(n)}_{k,m}$. The formula can be described as:

$$s^{(n)}_{k,m-1} = s^{(n)}_{k,m} + B_m \tag{18}$$

where B_m is the multi-variate Gaussian stochastic variation. Variance is P_m and mean is 0.
(5) $S_{k,m-1}$ which has been generated is used to initialize the $m - 1$ layer. The process is repeated until we arrive at the set $S^{\pi}_{k,0}$.
(6) $S^{\pi}_{k,0}$ is used to evaluate the optimal model configuration X_k using

$$X_k = \sum_{i=1}^{N} s^{(i)}_{k,0}\pi^{(i)}_{k,0} \tag{19}$$

(7) Then, the set $S_{k+1,m}$ is produced from $S^{\pi}_{k,0}$ using

$$s^{(n)}_{k+1,M} = s^{(n)}_{k,0} + B_0 \tag{20}$$

3.3 Annealing Rate of Tracking Parameter Setting

As stated previously the function $w_M(Z, X)$, used in each layer of the annealing process, is determined by Eq. 16 with $\beta_0 > \beta_1 > \cdots > \beta_M$. The value of β_m will determine the rate of annealing at each layer. A large β_m will produce a peaked weighting function w_m resulting in a high rate of annealing. Small values of β_m will have the opposite effect. If the rate of annealing is too high the influence of local maxima will distort the estimate of X_k. If the rate is too low X_k will not be determined with enough resolution.

A good measure of the effective number of particles will be chosen for next layer propagation. We do not use the exact gradient descent method, but the survival rate is adjusted by using annealing rate. It is a simple amendment on the basis of present survival rate, so the algorithm is simple and effective.

$$\beta(l) = \beta(l-1) - \varepsilon(\alpha_{target} - a(l-1)) \tag{21}$$

Where a_{target} is the expected survival rate of particles on each layer. $a(l-1)$ is the survival rate weighted on last layer. ε is the learning factor. It is usually set to $\frac{1}{l+1}$ and satisfied $\beta(l) \geq \beta(l-1)$.

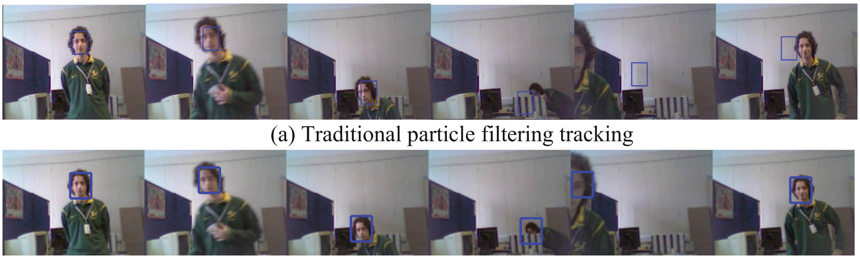
4 Experiment Results and Analysis

In order to verify the results of tracking method, a large amount of experiments is carried out. The sequences in experiments can be accessed human face tracking sequences [16, 17] Experiments are done on Pentium 2.4 GHZ CPU Common Configuration Computer. Image size is 320 × 240. Video capture rate is 25 frames per second. The initial state of particle region is set to $x_0 \sim U(0,320), y_0 \sim U(0,240), \Delta x \sim N(0,2)$, $\Delta y \sim N(-10,6)$ and the number of particles is $Ns = 100$. Annealing layers $m = 3$, $T_0 = 1000$, $T_{\min} = 50$, intex_max = 10, $\beta(l) = 0.99$, $a_0 = a_1 = a_2 = a_3 = 0.5$.

4.1 Real Time Tracking Experiment

The comparison of two tracking algorithms is given. In the first experiment, sequences one, which is constructed from the 84th frame to the 95th frame, experienced the procedure of face being sheltered and light becoming darker. From the 115th frame to the 130th frame, there was a fast squat downward procedure. From the 421th frame to the 430th frame, it's a full occlusion process. From the results of the experimental sequences, traditional particle filter tracking was failed. This is because it only uses color information. Then, when the target entered illumination area, the colors of the target changed dramatically, and the proposal distribution didn't utilize present observation information. Then, it could not capture color variation and lead the algorithm to fail. Current color, edge, and texture observation information can be utilized to the proposal distribution, so the target is well discriminated. Besides, edge information is not sensitive to the illumination variations. Thus, when the color feature lost its discrimination, edge and texture information played a leading role. It made the whole sequence reliably track the target. From the results of the two experiments, tracking of annealing particle filtering is effective and stable because the improvement of proposal distribution under complicated backgrounds situations, existing partial occlusion and similar target color (Figs. 3 and 4).

Second experimental sequences are selected from http://www.ces.clemson.edu/~stb/research/headtracker/. The tracked target is in a more complex tracking environment with the similar background, light, rotation, occlusion and background interference. Because the traditional particle filter can not achieve tracking, the paper does not give the experimental results. And the algorithm can achieve tracking precision.



(a) Traditional particle filtering tracking

(b) Proposed particle filtering tracking

Fig. 3. The first experiment results (1th, 107th, 184th, 432th, 537th,600th tracking results)

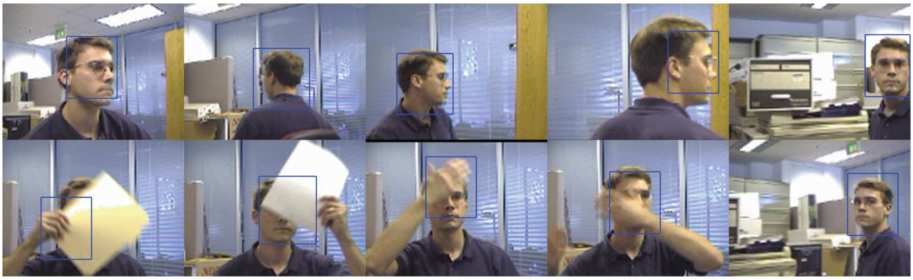


Fig. 4. The second experiment results (the 1th, 30th,47th, 89th, 203th, 233th, 247th, 287th, 296th, 445th tracking results)

4.2 Performance Analysis

4.2.1 Tracking Precision

In order to better describe the performance of the improved algorithm, the latest improvements of the other PF algorithms are compared with the algorithm of this paper for the number of different particles, the results are shown in Fig. 5 and Table 1. The average root mean square error (RMSE) of the SAPF algorithm is significantly lower than that of the PF algorithm under the same conditions. The average RMSE value of the SAPF algorithm decreases significantly with the increase of the number of particles. It shows the probability of the optimal estimation of the tracking approach. At the same time, it can be seen that if the PF algorithm achieves the same tracking accuracy of that of the SAPF algorithm, we must increase the number of particles, which leads to a long run time, can not meet the needs of real-time target tracking

4.2.2 Test Tracking Speed

The total running time of the PF and SAPF algorithm was observed at 100 times one time, and the results are shown in Table 2. From Table 2, it is easy to see that the importance sampling step of the improved algorithm is slightly complicated, the time consumption of the algorithm is slightly longer than that of the common PF algorithm. But the increased time overhead is within the scope of acceptance. With the

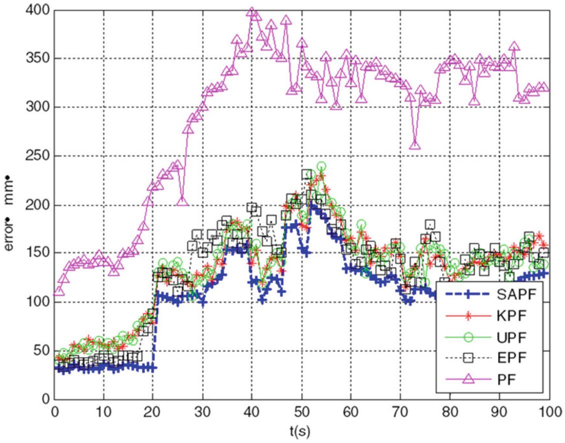


Fig. 5. RMSE of target tracking of PF, KPF, EPF, UPS, SAPF algorithms

Table 1. Average *RMSE* comparison results of PF, SAPF algorithms

RMSE	N = 50	N = 100
PF	12.5438	10.4623
SAPF	7.7842	6.3597

Note: 1 independent simulation

Table 2. Comparison results of the running time of PF, SAPF algorithms

TIME	N = 50	N = 100
PF	7.3250	12.2507
SAPF	7.9715	13.1782

Note: 1 independent simulation

improvement of the tracking accuracy, it can be concluded that the improved algorithm significantly improves the tracking performance of the system.

4.2.3 Comparison of Various Algorithms

Compared with other improved PF algorithms, the results are shown in Table 3.

Table 3. Performance comparison of PF, KPF, EPF, UPF, SAPF algorithm Unit: s

Algorithms	PF	KPF	EPF	UPF	SAPF
Average *RMSE*	9.2760	5.0542	8.6535	6.5428	3.6949
Average *TIME*	7.8158	9.7621	8.5125	12.0746	8.3826

Note: 50 independent simulations, N = 50

We can observe from Table 3 that the average RMSE values of the improved PF algorithm were closer to the true value than the PF algorithm. It indicates that the tracking accuracy and reliability of the improved PF algorithm are better than that of the basic algorithm of PF. But the time cost of the KPF, EPF and the UPF algorithm is difficult to adapt to the real-time target tracking. SAPF algorithm gradually made the tracking accuracy approach to the optimum by improving the importance sampling density function. The computing time is also able to meet the time constraints of the real-time target tracking.

5 Conclusion

The paper presents a kind of multi-feature integration annealing particle filtering method. Weighting function is produced by applying the image feature properties of the fusion between colors and edges to weight in different annealing layers, so the proposed distribution is improved. It not only makes the target never excessively depends on the individual feature, but also effectively solves the particle degradation problem. Algorithm thus can effectively avoid illumination variation and complicated backgrounds influence and our method obtains well tracking results. The future work will increase the contour information and texture information, and so on. We will pay more attention on the effectiveness of integration method.

Acknowledgments. This research was supported by the project of research foundation of the talent of scientific and technical innovation of Harbin City (NO. 2014RFQXJ103).

References

1. Doucet, A., de Freitas, N., Gordon, N. (eds.): Sequential Monte Carlo Methods in Practice. Springer, New York (2001)
2. Maggio, E., Cavallaro, A.: Hybrid Particle filter and mean shift tracker with adaptive transition model. In: IEEE International Conference on Acoustics (2005)
3. Xin, Y., Jia, L., Pengyu, Z.: Adaptive particle filter for object tracking based on fusing multiple features. J. Jilin Univ. **45**(2), 533–539 (2015). (Engineering and Technology Edition)
4. Gu, X., Wang, H., Wang, L.: Fusing multiple features for object tracking based on uncertainty measurement. Acta Autom. Sin. **37**(5), 550–559 (2011)
5. Maggio, E.: Adaptive multi-feature tracking in a particle filtering framework. IEEE Trans. Circuits Syst. Video Technol. **17**(10), 1348–1359 (2007)
6. Xiaowei, Z.: Particle filter tracking algorithm combining the color and structural information. Opto Electron. Eng. **35**(10), 1–6 (2008)
7. Deutscher, J., Blake, A., Reid, I.: Articulated body motion capture by annealed particle filtering. In: Proceedings of Conference on Computer Vision and Pattern Recognition, Hilton Head, South Carolina, vol. 2, pp. 1144–1149 (2000)
8. Yang, S., Wu, T., Zhang, Y.: Particle filter based on simulated annealing for target tracking. J. Optoelectron. Laser **22**(8), 1236–1240 (2011)

9. Li, Y., Sun, Z., Chen, S.: 3D human pose analysis from monocular video by simulated annealed particle swarm optimization. Acta Autom. Sin. **38**(5), 732–741 (2012)
10. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-based object tracking. IEEE Trans. Pattern Anal. Mach. Intell. **25**(5), 564–577 (2003)
11. Nummiaro, K., Koller-Meier, E., Gool, L.V.: An adaptive color-based particle filter. Image Vis. Comput. **21**, 99–110 (2003)
12. Pérez, P., Hue, C., Vermaak, J., Gangnet, M.: Color-based probabilistic tracking. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2350, pp. 661–675. Springer, Heidelberg (2002). doi:10.1007/3-540-47969-4_44
13. MacCormick, J.: Probabilistic modeling and stochastic algorithms for visual localization and tracking. Ph.D. thesis, University of Oxford, UK (2000)
14. López Méndez, A.: Feature-Based Annealing Particle Filter for Robust Motion Capture. Image and Video Processing Group, pp. 1–72 (2009)
15. Shao, P., Wan, C.: Genetic-annealing algorithm for global optimization problems. Comput. Eng. Appl. **43**(12), 62–65 (2007)
16. http://www.ces.clemson.edu/ ~ stb/research/headtracker/
17. ftp://motinas.elec.9mul.ac.uk/pub/single-face

Reflecting on Imbalance Data Issue When Teaching Performance Measures

Pavel Škrabánek(✉) and Filip Majerík

Faculty of Electrical Engineering and Informatics, University of Pardubice,
Studentská 95, 532 10 Pardubice, Czech Republic
pavel.skrabanek@upce.cz

Abstract. Importance of soft computing methods has continuously grown for many years. Particularly machine learning methods have been paid considerable attention in the business sphere and subsequently within the general public in the last decade. Machine learning and its implementation is the object of interest of many commercial subjects, whether they are small companies or large corporations. Consequently, well-educated experts in the area of machine learning are highly sought after on the job market. Most of the technical universities around the world have incorporated the machine learning into their curricula. However, machine learning is a dynamically evolving area and the curricula should be continuously updated. This paper is intended to support this process. Namely, an imbalance data issue, in context of performance measures for binary classification, is opened, and a teaching method covering this problem is presented. The method has been primary designed for undergraduate and graduate students of technical fields; however, it can be easily adopted in curricula of other fields of study, e.g. medicine, economics, or social sciences.

Keywords: Machine learning · Binary classification · Imbalanced data · Performance measures · Teaching method

1 Introduction

At present, the term 'soft computing' is widely used in different contexts. Indeed, it covers a number of different methods, e.g. evolutionary computation, neural computing, or machine learning methods. The soft computing methods are widely studied in academia; however, many commercial companies fund their own research and development. Especially, machine learning has been paid considerable attention in the business sphere in the last decade. In this context companies, for example, Google Inc., Microsoft Corporation, or Facebook Inc. are probably the best known.

Machine learning methods are already employed in many commercial products and the number continuously grows. Accomplishment of the already introduced solutions naturally attracts attention of other companies. Consequently, well-educated experts in the area of machine learning are highly required on the job market.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_4

Most of the technical universities around the world have incorporated machine learning into their curricula. The curricula usually cover essential theory; however, some topics, important in terms of practice, are omitted. Let us consider only classification tasks. As experienced practitioners know well, quality of a classifier depends on many aspects. For example, Salzberg [21,22] mentions selection of classification algorithms, and forming of training and test sets in this context. However, Japkowicz [14] notes that correct evaluation of a classifier's performance is fundamental.

The performance is assessed using a *performance measure*. A good overview of the measures can be found in [12]. Unfortunately, student and novice understanding of the measures seems to be insufficient. Indeed, the purely motivated way of using performance measures and using of incompatible methods has been criticized in many papers [8,14,20]. This topic has attracted adequate attention of researchers [13,14,23] which has been positively reflected in current thematic books [7,17]. However, there is one important issue which is still neglected in the books, and unfortunately, sometimes during teaching as well.

The marginalized topic is an issue of *imbalanced data* (also known as class imbalance or skewed class distributions problem) and its influence on the performance measures [15]. To be fair, existence of the problem is mentioned in the current books; however, no extra attention is paid to description of a context. Nevertheless, our practical experience shows that discussion of its roots helps for a better understanding of the measures by students which is positively reflected further in the classroom. Moreover, importance of this topic is evident from a long queue of publications about the imbalanced data and related issues. Besides the already mentioned, let us refer at least to [1,5,9,18].

Considering all these facts, the influence of the imbalanced data on the performance measures should be integrated to the curricula. We have proved the binary classification to be optimal for this task; thus, we have based our teaching method on performance measures for the binary classification. We wish to share the method among the people dealing with machine learning, and thereby contribute to improving the current state of art.

The rest of the paper is organized in the following way. Basic variables used by evaluation of binary classifiers and the most popular performance measures are introduced in Sect. 2. The influence of imbalanced data on the performance measures is analyzed in Sect. 3. Approaches used by handling of the imbalanced data problem are summarized in Sect. 4. A proposed teaching method is presented in Sect. 5. The conclusion is stated in Sect. 6.

2 Performance Measures Used by Evaluation of Binary Classifiers

Two classes, *positive* or simply P, and *negative* or simply N, are considered by the binary classification. The aim of a classifier is to correctly assign a class label to each judged sample. For each sample, the decision-making process falls into one of four possible scenarios: the sample is positive and the classifier correctly recognizes it as such (*True Positive* or simply TP); the sample is negative and the classifier correctly recognizes it as such (*True Negative* or simply TN); the sample is positive but the classifier labels it as negative (*False Negative* or simply FN); or the sample is negative but the classifier labels it as positive (*False Positive* or simply FP).

On the base of the presented scenarios, four fundamental quantities for performance measure are formulized: number of true positive $|TP|$; number of true negative $|TN|$; number of false negative $|FN|$; and number of false positive $|FP|$ samples. The quantities are usually summarized into a 2×2 matrix. The matrix is known as *confusion matrix* and it is traditionally expressed as in Table 1.

Table 1. The confusion matrix

		Assigned label	
		positive	negative
True label	positive	$\lvert TP\rvert$	$\lvert FN\rvert$
	negative	$\lvert FP\rvert$	$\lvert TN\rvert$

A number of performance measures derived from the confusion matrix have been introduced up to the present [4]. However, not all of them have been widely accepted. Moreover, different measures are preferred in various scientific fields. Thus, only the most frequently used measures are considered further. Namely, following measures are considered: accuracy (acc), error rate (er), precision (pr), recall (re), specificity (sp), false negative rate (fnr), false positive rate (fpr), harmonic mean of precision and recall (Fscore), geometric mean of precision and recall (Gmean), and area under the ROC[1] curve (AUC).

The enumerated measures are detailed in Table 2. Acronyms of the measures used in this paper are stated in the first column. The usual formulae are listed in the second one. The last column contains their modified expressions where the used notation is explained later in Sect. 3.

3 Influence of Imbalanced Data on Performance Measures

The imbalance in data is caused by a non-uniform distribution of classes in a set of labeled samples. The unfavorable properties of some performance measures on imbalanced data have been known for a long time; however, a publication clearly and simply explaining their essence is hard to find. In this section, the relation between the measures and proportions of the classes in test sets is analyzed. On the basis of the analysis, all the measures listed in Table 2 are expressed in the terms used within the analysis. The modified expressions are summarized in the third column of Table 2.

[1] ROC = receiver operating characteristic.

Table 2. The most popular performance measures in the binary classification

Acronym	Standard expression	Modified expression
acc	$\frac{\lvert TP\rvert+\lvert TN\rvert}{\lvert TP\rvert+\lvert FN\rvert+\lvert TN\rvert+\lvert FP\rvert}$	$\xi_{TP}.\nu_P + \xi_{TN}.\nu_N$
er	$\frac{\lvert FP\rvert+\lvert FN\rvert}{\lvert TP\rvert+\lvert FN\rvert+\lvert TN\rvert+\lvert FP\rvert}$	$(1-\xi_{TP}).\nu_P + (1-\xi_{TN}).\nu_N$
pr	$\frac{\lvert TP\rvert}{\lvert TP\rvert+\lvert FP\rvert}$	$\frac{\xi_{TP}.\nu_P}{\xi_{TP}.\nu_P+(1-\xi_{TN}).\nu_N}$
re	$\frac{\lvert TP\rvert}{\lvert TP\rvert+\lvert FN\rvert}$	ξ_{TP}
sp	$\frac{\lvert TN\rvert}{\lvert TN\rvert+\lvert FP\rvert}$	ξ_{TN}
fnr	$\frac{\lvert FN\rvert}{\lvert TP\rvert+\lvert FN\rvert}$	$1-\xi_{TP}$
fpr	$\frac{\lvert FP\rvert}{\lvert TN\rvert+\lvert FP\rvert}$	$1-\xi_{TN}$
Fscore	$\frac{(\beta^2+1).\lvert TP\rvert}{(\beta^2+1).\lvert TP\rvert+\beta^2.\lvert FN\rvert+\lvert FP\rvert}$	$\frac{(\beta^2+1).\xi_{TP}.\nu_P}{[(\beta^2+1).\xi_{TP}+\beta^2.(1-\xi_{TP})].\nu_P+(1-\xi_{TN}).\nu_N}$
Gmean	$\sqrt{\frac{\lvert TP\rvert}{\lvert TP\rvert+\lvert FN\rvert} \times \frac{\lvert TN\rvert}{\lvert TN\rvert+\lvert FP\rvert}}$	$\sqrt{\xi_{TP} \times \xi_{TN}}$
AUC	$\frac{1}{2}\left(\frac{\lvert TP\rvert}{\lvert TP\rvert+\lvert FN\rvert} + \frac{\lvert TN\rvert}{\lvert TN\rvert+\lvert FP\rvert}\right)$	$\frac{1}{2}(\xi_{TP}+\xi_{TN})$

3.1 Preliminary

Let us consider a test set of M labeled samples where each sample belongs either to the class P or N then

$$M = |P| + |N|, \tag{1}$$

where $|P|$ is number of positive samples, and $|N|$ is number of negative samples in the test set.

Supposing the confusion matrix stated in Table 1, the numbers of samples belonging to the classes can be expressed as

$$|P| = |TP| + |FN|, \tag{2a}$$

$$|N| = |TN| + |FP|, \tag{2b}$$

which allow us to express (1) as

$$M = |TP| + |FN| + |TN| + |FP|. \tag{3}$$

Let us express the numbers of samples in the classes as

$$|P| = \nu_P.M, \tag{4a}$$

$$|N| = \nu_N.M, \tag{4b}$$

where ν_P is the proportion of the positive samples in the test set, and ν_N is the proportion of the negative samples in the test set. Furthermore, it holds that $\nu_P, \nu_N \in [0, 1]$ and $\nu_P + \nu_N = 1$.

3.2 Analysis

Let us consider the objective of a binary classifier now. As was stated in Sect. 2, the aim of the classifier is to correctly assign a sample to one of the two classes, P or N, if possible. A well working classifier will correctly assign all the samples, i.e. $|TP| = |P|$, $|TN| = |N|$, $|FP| = 0$, and $|FN| = 0$. A classifier with a worse performance will correctly classify a smaller proportion of the samples. Thus let us express the number of correctly classified samples as

$$|TP| = \xi_{TP}.|P|, \tag{5a}$$

$$|TN| = \xi_{TN}.|N|, \tag{5b}$$

where ξ_{TP} is the proportion of correctly classified samples from all positive samples in the test set, ξ_{TN} is the proportion of correctly classified samples from all negative samples in the test set, and $\xi_{TP}, \xi_{TN} \in [0, 1]$.

On the basis of Eqs. (2) and (5), the numbers of miss-classified samples can be expressed as

$$|FN| = (1 - \xi_{TP}).|P|, \tag{6a}$$

$$|FP| = (1 - \xi_{TN}).|N|. \tag{6b}$$

It is obvious that performance of a binary classifier can be positively determined using only two quantities, ξ_{TP} and ξ_{TN}.

Now, let us explain the modification of the standard expressions which are stated in the second column of Table 2. The modification is demonstrated on the accuracy. Using the Eqs. (3) and (5), the original formulation can be expressed as

$$\text{acc} = \frac{\xi_{TP}.|P| + \xi_{TN}.|N|}{M}. \tag{7}$$

This equation can be further modified using (4), i.e.

$$\text{acc} = \frac{\xi_{TP}.\nu_P.M + \xi_{TN}.\nu_N.M}{M} = \xi_{TP}.\nu_P + \xi_{TN}.\nu_N. \tag{8}$$

From this equation, it is clearly evident that the accuracy does not depend only on performance of a classifier (ξ_{TP} and ξ_{TN}); however, composition of a test set is also reflected in this measure (ν_P and ν_N). Only for the sake of completeness, the accuracy, as well as the other measures based on the confusion matrix, is also influenced by the total number of samples in the test set M, and by selection of the samples. Since this is not significant for the topic covered in this paper, we refer to [17] for more information.

It is obvious that the same procedure can be applied on other measures based on the confusion matrix. As was already mentioned, modified expressions of the most popular measures are stated in the third column of Table 2. The formulas clearly show whether a measure is influenced by the test set composition, or not. It is apparent that besides the accuracy and the error rate, the precision and the Fscore are influenced by the test set composition; however, the others are invariant.

4 Handling Imbalanced Data

As pointed out by García et al. [10], two types of approaches can be used to handle the class imbalance data by evaluation of binary classifiers. Namely, a re-sampling method can be used, or measuring of a classifier's performance in imbalanced domains can be performed.

The re-sampling methods include many different approaches, such as, random or focused over-sampling [15]; over-sampling with informed generation of new samples [3]; random under-sampling [16]; or direct under-sampling [19].

As was demonstrated in Sect. 3, some measures are not influenced by the distribution of classes in test sets. García et al. [10] pointed to this fact and they have logically inferred that these measures can be safely used on imbalanced data. However, modifications of the biased measures, which are resistant to the imbalance, have been introduced, too [2,24]. Of course, ROC graphs should not be omitted here. A detailed summary about them can be found in [6].

5 Teaching Method

The proposed teaching method responsively complements the curricula. It covers the imbalanced data issue in context of the performance measures. The modification is based on the manner of their expression which was presented in Sect. 3 (see modified expression in Table 2). The suggested workflow of the method is summarized in Table 3.

The sequence numbers of the steps are in the first column. Communication between a teacher and students is apparent from the second one, where T denotes the teacher, and S is used for students. Modes of communication are symbolized using arrows. Their orientations express direction of the communication, i.e. T $\rightarrow$ S means that the teacher speaks to students, T $\leftarrow$ S symbolizes an independent activity of students leading to a solution of a given task, and T $\leftrightarrows$ S is used for discussion between the teacher and students. Description of activities is stated in the third column. Variables introduced in particular steps are in the last column.

The first step of the workflow is a general introduction where the objective of binary classifiers is explained and the concept of labeled samples, including specification of the classes, is introduced. It does not diverge fundamentally from the current practice and it is not necessary to cover the domain immediately prior to step two. In fact, it can be lectured even sooner, once a structure of a lecture requires that.

The second step departs from the routine. After the idea of an evaluation process is explained to students and basic notation related to the test set is introduced, students are asked to propose a method of classifier evaluation. Under guidance of the teacher, two most intuitive measures, ξ_{TP} and ξ_{TN}, are introduced. Active participation of students in this step is one of the critical points of the teaching method. The level of their involvement in this moment affects their further progress.

Since the workload of the next two steps is evident from Table 2, it is not described in detail here. All necessary facts can be found in Sect. 2. In the fifth

Table 3. Workflow by lecturing of performance measures for binary classification

Step	Acting	Description of activities	Introduced variables
1	T → S	Explanation of the purpose of a binary classification, introduction of the concept of labeled samples and its use	P, N
2	T ⇆ S	Introduction of test sets and design of two most intuitive measures for evaluation of binary classifiers	M, $\lvert P\rvert$, $\lvert N\rvert$, ξ_{TP}, ξ_{TN}
3	T → S	Analysis of possible outcomes by the classification process	TP, TN, FP, FN
4	T → S	Introduction of the confusion matrix and of the related quantities	$\lvert TP\rvert$, $\lvert TN\rvert$, $\lvert FP\rvert$, $\lvert FN\rvert$
5	T → S	Introduction of selected performance measures	acc, er, pr, sp, ...
6	T ⇆ S	Question stated by teacher: 'Does the proportion of classes in a test set influence performance measures?'	
7	T → S	Expression of test sets compositions using the proportions	ν_P, ν_N
8	T ← S	Task for students: 'Modify standard formulas of two given measures using $\xi_{TP}, \xi_{TN}, \nu_P, \nu_N$.' Note: One of the measures should be dependent on the composition, e.g. the accuracy, and one should be independent, e.g. the recall. More details can be found in Sect. 3	
9	T → S	Check of student's results and introduction of modified expression for remaining considered measures	
10	T ⇆ S	Discussion about the measures and their biasing by the test set composition considering the modified expressions (see fourth column of Table 2)	
11	T → S	Introduction of solutions allowing evaluation on imbalanced data (see Sect. 4)	

step, students are familiarized with the most important measures. Although we present our selection of the most important measures in Table 2, their choice can be adapted according to specific requirements of a particular curriculum. In the sixth step, the issue of the imbalanced data is opened. The teacher shows expression of the test set composition using the proportions ν_P and ν_N. After that, the most critical step comes.

The most critical step is the eighth one. The level of understanding of students to the issue is highly dependent on the degree of mastering of the given task.

The students are asked to express two selected measures using the proportions of the classes in the test set, ν_P and ν_N; and using the proportions of correctly classified samples, ξ_{TP} and ξ_{TN}. The way of derivation is described in Sect. 3. For that purpose, one biased and one unbiased measure should be used. The contrast between them supports the understanding.

Results of the student's independent work are checked by the teacher in the ninth step. Subsequently, the teacher introduces modified expressions of the remaining measures. At this moment, students should clearly see whether a measure is biased or not. This allows opening of a discussion about consequences of this phenomenon which is the purpose of the tenth step.

The last step given in Table 3 suggests building on the opened issue by introduction of solutions covering the drawback of biased measures (see Sect. 4). However, another appropriate topic can be discussed instead of it, e.g. probabilistic interpretation of the measures in the context of imbalanced data [2,11], or measures for multi-class classification in context of class distribution [23]. We keep this step as an open question where each teacher can choose the most suitable topic. Naturally, the choice of the topic should reflect the plan of a particular lecture.

6 Conclusion

Importance of soft computing methods is unfailingly growing. Since soft computing is a constantly evolving field, the curricula should be continuously updated. In this paper, the imbalanced data issue related to classification tasks was opened and a teaching method covering this issue was suggested. Although the issue has been known for a long time, it is often overlooked. The consequences of such an approach are evident in practice which has been repeatedly criticized [8,14,20].

The teaching method introduced in this paper aims to cover this deficiency. Although its implementation does not require an extensive modification of current curricula, its benefits are expressive. In addition to complementation of a student's knowledge about the imbalanced data issue, its application significantly contributes to understanding of the measures by students. We expect that the substance of the better understanding consists in active involvement of students with high emphasis on their creativity.

The presented method has been primary designed for undergraduate and graduate students of technical fields. Since the topic 'performance measure' usually immediately follows the basic machine learning theory in the same course, no special prior knowledge of the students, beyond the current requirements, is necessary. Indeed, the method has been verified within our course aimed on introduction to artificial intelligence which is lectured to graduate students. The method has proven to be efficient in our classes. We believe that the positive results in the classroom will be reflected in practice soon.

Although the method has been primary designed for technical field of study, it can be easily adopted in other fields, e.g. in medicine, in economics or in social sciences. We hope that the idea described in this paper will be widely accepted, and thereby contribute to improvement of the education system.

Acknowledgement. The work has been supported by the Funds of University of Pardubice, Czech Republic, grant No. SGS-2016-017. This support is very gratefully acknowledged.

References

1. Barandela, R., Sánchez, J.S., García, V., Rangel, E.: Strategies for learning in class imbalance problems. Pattern Recogn. **36**(3), 849–851 (2003)
2. Brodersen, K., Ong, C.S., Stephan, K., Buhmann, J.: The balanced accuracy and its posterior distribution. In: 2010 20th International Conference on Pattern Recognition (ICPR), pp. 3121–3124, August 2010
3. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: synthetic minority over-sampling technique. J. Artif. Intell. Res. **16**(1), 321–357 (2002)
4. Choi, S.S., Cha, S.H., Tappert, C.: A survey of binary similarity and distance measures. J. Systemics Cybern. Inform. **8**(1), 43–48 (2010)
5. Daskalaki, S., Kopanas, I., Avouris, N.M.: Evaluation of classifiers for an uneven class distribution problem. Appl. Artif. Intell. **20**(5), 381–417 (2006)
6. Fawcett, T.: An introduction to ROC analysis. Pattern Recogn. Lett. **27**(8), 861–874 (2006)
7. Flach, P.: Machine Learning: The Art and Science of Algorithms that Make Sense of Data. Cambridge University Press, New York (2012)
8. Forman, G., Scholz, M.: Apples-to-apples in cross-validation studies: pitfalls in classier performance measurement. SIGKDD Explor. **12**(1), 49–57 (2010)
9. Garcia, V., Mollineda, R.A., Sanchez, J.S.: Theoretical analysis of a performance measure for imbalanced data. In: 2010 20th International Conference on Pattern Recognition, pp. 617–620. IEEE, August 2010
10. García, V., Sánchez, J.S., Mollineda, R.A., Alejo, R., Sotoca, J.M.: The class imbalance problem in pattern classification and learning. In: Ferrer-Troyano, F.J., et al. (eds.) II Congreso Español de Informática, pp. 283–291. Thomson, Zaragoza (2007)
11. Goutte, C., Gaussier, E.: A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. In: Losada, D.E., Fernández-Luna, J.M. (eds.) ECIR 2005. LNCS, vol. 3408, pp. 345–359. Springer, Heidelberg (2005). doi:10.1007/978-3-540-31865-1_25
12. Hand, D.J.: Assessing the performance of classification methods. Int. Stat. Rev. **80**(3), 400–414 (2012)
13. Hossin, M., Sulaiman, M.: A review on evaluation metrics for data classification evaluations. Int. J. Data Min. Knowl. Manage. Process **5**(2), 1–11 (2015)
14. Japkowicz, N.: Why question machine learning evaluation methods. In: AAAI Workshop on Evaluation Methods for Machine Learning, pp. 6–11 (2006)
15. Japkowicz, N., Stephen, S.: The class imbalance problem: a systematic study. Intell. Data Anal. **6**(5), 429–449 (2002)
16. Kotsiantis, S., Pintelas, P.: Mixture of expert agents for handling imbalanced data sets. Ann. Math. Comput. Teleinformatics **1**(1), 46–55 (2003)
17. Kubat, M.: An Introduction to Machine Learning. Springer, Cham (2015)
18. Lemnaru, C., Potolea, R.: Imbalanced classification problems: systematic study, issues and best practices. In: Zhang, R., Zhang, J., Zhang, Z., Filipe, J., Cordeiro, J. (eds.) ICEIS 2011. LNBIP, vol. 102, pp. 35–50. Springer, Heidelberg (2012). doi:10.1007/978-3-642-29958-2_3

19. Mani, I., Zhang, I.: kNN approach to unbalanced data distributions: a case study involving information extraction. In: Proceedings of Workshop on Learning from Imbalanced Datasets (2003)
20. Powers, D.M.W.: Evaluation: From precision, recall and f-measure to ROC, informedness, markedness & correlation. J. Mach. Learn. Technol. **2**(1), 37–63 (2011)
21. Salzberg, S.L.: On comparing classifiers: a critique of current research and methods. Data Min. Knowl. Disc. **1**, 1–12 (1999)
22. Salzberg, S.L.: On comparing classifiers: pitfalls to avoid and a recommended approach. Data Min. Knowl. Discov. **1**(3), 317–328 (1997)
23. Sokolova, M., Lapalme, G.: A systematic analysis of performance measures for classification tasks. Inf. Process. Manage. **45**(4), 427–437 (2009)
24. Velez, D.R., White, B.C., Motsinger, A.A., Bush, W.S., Ritchie, M.D., Williams, S.M., Moore, J.H.: A balanced accuracy function for epistasis modeling in imbalanced datasets using multifactor dimensionality reduction. Genet. Epidemiol. **31**(4), 306–315 (2007)

Ontology Based Knowledge Representation for Cognitive Decision Making in Teaching Electrical Motor Concepts

Atul Prakash Prajapati(✉) and D.K. Chaturvedi

Faculty of Engineering, DayalBagh Educational Institute, Agra 282005, UP, India
atulprakash21@gmail.com, dkc.foe@gmail.com
http://www.dei.ac.in

Abstract. Background. Cognitive decision-making is a promising area of research. As the word, cognitive connote the human-like behaviour, and humans acquire their knowledge based on their day to day experience. To accomplish this proficiency our agent or system should behave astutely. Here the role of Knowledge representation comes into the picture, and only the best representation tool can fabricate the domain knowledge efficiently. In this paper, we present the ontological aspect to fabricate the domain knowledge for effective decision-making.

Objective. The objective of this paper is to exhibit the role of ontology for designing an effective knowledge representation system, which leads towards an effectual cognitive decision-making system.

Methodology. This paper selects a particular area "Electrical Motor" and builds up an ontology to demonstrate basic concepts related to the electric motor. This ontology is useful for answering the questions related to basic motor concepts; that is useful to educate the students.

Result. This paper deals with the realistic aspects related to the design of an ontology. It converses the anticipated experimental design issues and results.

Conclusion. Finally this paper concludes the ontology development process by describing the pros and cons. It also converses the future aspects (NLP Modelling) for knowledge representation, which can effectively model a domain knowledge.

Keywords: Ontology · Knowledge representation techniques · Cognitive decision making

1 Introduction

With the development of technology, there is an enormous need for intelligent expert systems that act astutely in indefinite conditions and environments. It leads to the obligation of an effective decision-making capability within the agent. Decision-making competence comes with the reasoning and inferencing power for

A.P. Prajapati—is working in the area of "Cognitive Computing".

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_5

that knowledge base is requisite. So for the advancement of the system's performance, knowledge management systems play a vital role. Various knowledge modelling and management systems have designed in this area. The following section presents a brief survey of the ontology-based systems.

Traditional GIS (Graphic information system) systems provide a limited set of activity like emergency response and management. To improvise the use of these GIS systems, Lee proposed an android OS based GIS system. It has 2-D and 3-D visualization and LBS (Location based system) system that uses ontological querying to track and manage indoor activities in a university [1]. Liu proposed an ontological approach for representing domain knowledge in remote sensing applications. Here the domain ontology is constructed by using Protégé tool [2]. Raies explained that designing SGBLS (Smart game based learning system) systems is an exigent task because it requires proficiency and handiness. So to improve the knowledge and dexterity in the neophyte designers, this paper proposed the importance of an ontology for semantic annotation [3]. Ontology is an approach to imply knowledge in semantic web, Liu proposed an ontological way to represent fuzzy knowledge using cloud models and fuzzy set methods [4]. Morente proposed a fuzzy ontology-based decision support system. He developed an android application having a web platform that has a decision-making capability using mobile devices. It helps the users to choose a particular wine for their dinners [5]. Traditional clinical management systems cannot effectively deal with patient information, so Zhang proposed an ontological approach for modelling healthcare domain knowledge and patient data. As the ontology is capable of showing the relationship in between different entities in the domain, this improves the reasoning and decision-making capability. Here type-2 diabetic patient's data has taken as a case study for checking the reliability and system performance [6]. Chuprina proposed an ontology-based web service system "Reply" having NLP interfacing. This NLP interfacing provides an effortless and user-friendly environment for the neophyte users. This system converts an NLP-query to an equivalent SQL-query for accessing data from structured data sources [7]. Maffei proposed a comparative study in between constructivism theory and transmissive approach. This paper proposed constructivism based ontology "CONALI". It uses "OWL" as a modelling language [8]. Chuprina illustrated that, how do an ontology help students, industry specialists, and academic researchers? He suggested some key aspects that should be considered while designing an ontology [9]. Wang described the role of ontology modularization. It improves the Human-Web communication and also is an effective approach for knowledge sharing [10]. Tadjine pointed out the drawbacks present in the designing of traditional LMS (Learning Management Systems) systems. Designers use multimedia contents to improve the learning usually. In distinction, this paper focused on "Pattern-Based-Learning" that uses ontology model for designing an effective LMS systems [11]. Chun proposed a hybrid approach for designing cyber-security ontology. It contains domain knowledge as well as learning and teaching related knowledge also. The purpose of this approach was to improve the learning in students. It contains a digital textbook having an index for improving readability [12]. Nuntawong proposed a system to improve the course standards provided by the academies in Thailand. It consists of several ontologies for different courses and frameworks for higher education (TQF: HED). He developed a web application based system that maps two ontologies (Curriculum ontology Occ and Computer science TQF: HED ontology OTQF). Additionally, they used Wu and Palmer's algorithm for calculating the similarity between ontologies [13]. Lalingkar proposed an ontology based tutoring system "MONTO" for educating mathematical problems [14]. To overcome the problems present in the existing systems, Chi proposed an ontology-based approach for on-line self-regulated learners. This proposed ontology uses a triplet combination for representing knowledge (Guide, Material, and Teaching). It uses Protégé tool for creating mathematics curriculum ontology [15]. Altun explained the importance of ontology for designing object repositories. He proposed an ontology based learning and retrieval tool (OBELON). He also compared the performance of OBELON with a taxonomy based repository system and found that ontology-based systems are better than taxonomy based systems [16].

Section 1 of this paper presents a survey of systems designed in the field of ontology. Sections 2 and 3 describes the background and methodology respectively. Section 4 describes the experimental design set up for designing an effective decision support system. Finally, conclusion and future work suggest the augmentation that is possible in the existing system for cognitive decision-making.

2 Background

For knowledge representation and management, there subsist several tools and systems. Rule-based representation, graph-based representation, and semantic networks are some techniques that can efficiently embody a domain knowledge. Semantic network or concept network is similar to a graph based approach that shows the relationship in between concepts. We represent concepts in the form of triples (Subject, object, and predicate) all in the form of URI (Uniform resource identifier). URI is a superset of URL (Uniform resource locator, that provides the location of the resource on the Web)

URI = scheme "://" authority "/" path [query] [fragment].
authority = [userinfo "@"] host [":" port].

Analogously an ontology act like a dictionary for a particular domain. In ontology, we construct classes, class hierarchy and properties. Classes symbolize the entities and properties describe the association in between the entities (Classes and Individuals). Tools like "Protégé–5.1.0" developed by "$StanfordUniversity$" can sculpt an ontology effectively. Following Fig. 1 illustrates the architecture of an ontology development tool.

To signify electrical motor concepts, we introduce an Ontology $\alpha_{(ELE)}$, a Knowledge-Base module $KB_{(ELE)}$, and a Reasoner module $R_{(ELE)}$.

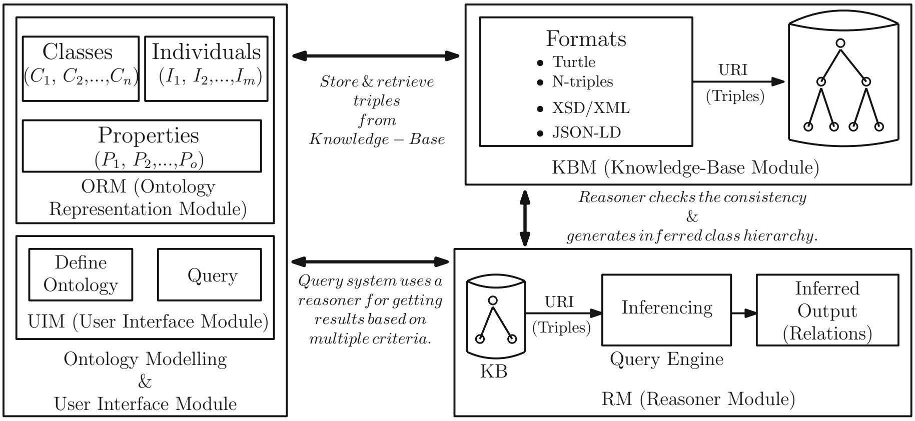

Classes
(C_1, C_2,...,C_n)
Individuals
(I_1, I_2,...,I_m)
Properties
(P_1, P_2,...,P_o)
ORM (Ontology Representation Module)
Define Ontology
Query
UIM (User Interface Module)
Ontology Modelling & User Interface Module
Store & retrieve triples from Knowledge – Base
Query system uses a reasoner for getting results based on multiple criteria.
Formats
Turtle
N-triples
XSD/XML
JSON-LD
URI (Triples)
KBM (Knowledge-Base Module)
Reasoner checks the consistency & generates inferred class hierarchy.
KB
URI (Triples)
Inferencing
Inferred Output (Relations)
Query Engine
RM (Reasoner Module)

Fig. 1. Schematic diagram showing an architecture of an Ontology

Definition 1. *A domain based Ontology $\alpha_{(ELE)}$ is a set having 4-tuples (C, P, I, A).*

$$C = \{(c_i), \forall i \in [1, \ldots, n]\} \tag{1}$$

C represents a set of classes. A class is a concept that contains a set of individuals. Here, we have a super-class, sub-class concept also. By default, all the elements of a sub-class belong to its super-class. Here, we used circle or oval to represent Classes.

$$P = \{(p_j), \forall j \in [1, \ldots, m]\} \tag{2}$$

P represents a set of properties. It is a binary relation in between individuals or classes. One can classify Property set into two categories (Object properties, Data properties).

$$I = \{(i_k), \forall k \in [1, \ldots, o]\} \tag{3}$$

I represent a set of individuals. Individuals are instances or objects of classes.

$$\begin{aligned} A = \{(p_j, i_k, i_{k+1}) \parallel (p_j, c_i, c_{i+1})\} \\ : \{\forall i \in [1, \ldots, n], \forall j \in [1, \ldots, m], \forall k \in [1, \ldots, o]\} \end{aligned} \tag{4}$$

A is a set of axioms.

Definition 2. *A knowledge-base $KB_{(ELE)}$ is domain specific. It has 3-tuples $\{T, R, A\}$.*

$$\begin{aligned} T = \{(t_i), \forall i \in [1, \ldots, n]\} \\ : \{t_i \equiv \langle S, \Pr, O\rangle, \forall \{S \in [Subject], \Pr \in [\Pr edicate], O \in [Object]\}\} \end{aligned} \tag{5}$$

T represents a set of a triples $\{t_i \equiv \langle S, Pr, O\rangle\}$. In knowledge-base, every piece of knowledge is represented as a statement of triple $\langle Subject, Predicate, Object\rangle$.

$$\left\{ \begin{array}{l} S \in [Subject]\,, \, is\, represented\, as\, either\, URI\, or\, Blank - Nodes. \\ Pr \in [Predicate]\,, \, is\, represented\, as\, URI\, only. \\ O \in [Object]\,, \, is\, represented\, as\, URI\, or\, Blank - Nodes. \end{array} \right\} \tag{6}$$

A triple $t_i \equiv \langle S, Pr, O\rangle$, has these constraints.

$$\begin{aligned} R &= \{(r_j)\,, \forall j \in [1, \ldots m]\} \\ &: \{(t_i, r_j, t_{i+1}) \equiv (t_i,\, is\, related\, to, t_{i+1},\, with\, relation\, r_j)\} \end{aligned} \tag{7}$$

R represents a set of relation; that relates entities.

$$\begin{aligned} A &= (t_i, r_j, t_{i+1}) \\ &: \{\forall i \in [1, \ldots, n]\,, \forall j \in [1, \ldots, m]\} \end{aligned} \tag{8}$$

A is a set of axioms.

Definition 3. *A reasoner $R_{(ELE)}$ has 2-tuples (A_c, I_c).*

$$\left\{ \begin{array}{l} (A_c \subset C)\,, \, is\, a\, set\, of\, classes\, called\, asserted\, class\, hierarchy. \\ It\, is\, an\, input\, to\, the\, reasoner. \\ (I_c \subset C)\,, \, is\, a\, set\, of\, classes\, called\, inferred\, class\, hierarchy. \\ It\, is\, an\, output\, from\, the\, reasoner. \end{array} \right\} \tag{9}$$

3 Method

This segment of the paper elaborates the step by step method for designing an ontology.

Algorithm 1 explains the principle of the reasoner. The reasoner module $R_{(1)}$ checks the consistency of the ontology and also computes the class hierarchy. Here, we have (A_c, I_c) for representing asserted class hierarchy and inferred class hierarchy respectively, $(c_i, c_j) \in C$ and (Sc_i, Sc_j) are subclasses of (c_i, c_j) respectively. If we have two instances of classes (c_i, c_j) that are disjoint, still they have some common properties, then the reasoner should produce an error message.

Following Fig. 2 describes the ontology development process. Initially, we collect information regarding the domain, create classes (C), subclasses, define properties (P); then the reasoner module $R_{(1)}$ produces the inferred class hierarchy (I_c) for checking the consistency of the ontology. If there is an inconsistency, it highlights them. Final outcome of all these steps is a domain based ontology (α_{Domain}).

Algorithm 1. Agorithm for Classification of Ontology

```
procedure CLASSIFY($A_c, I_c$)                                        ▷ (I/P and O/P)
    $Initilize\,(A_c)$
    if $((c_i \cap c_j = \phi)\ \&\&\ (\exists Sc_j \in c_i \parallel \exists Sc_i \in c_j))$ then   ▷ (Disjoint classes having
        common properties, (not allowed))
        $answer \leftarrow 0$
    else
        $answer \leftarrow 1$
    end if
    if $(answer)$ then
        $return(Ic)$                                                  ▷ (Inferred Class Hierarchy)
    else
        $Inconsistent\,Ontology$
    end if
    Return $(I_c)$                                                    ▷ Inferred class hierarchy
end procedure
```

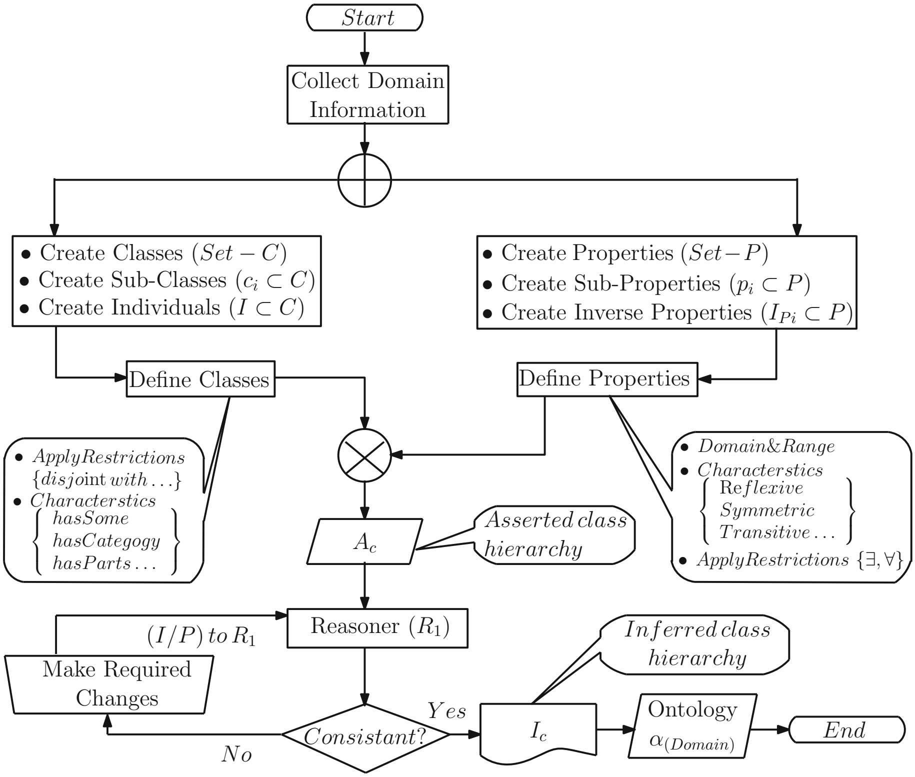

Start
Collect Domain Information
• Create Classes $(Set - C)$
• Create Sub-Classes $(c_i \subset C)$
• Create Individuals $(I \subset C)$
• Create Properties $(Set - P)$
• Create Sub-Properties $(p_i \subset P)$
• Create Inverse Properties $(I_{Pi} \subset P)$
Define Classes
Define Properties
• ApplyRestrictions {disjoint with ...}
• Characterstics {hasSome, hasCategogy, hasParts ...}
• Domain&Range
• Characterstics {Reflexive, Symmetric, Transitive ...}
• ApplyRestrictions $\{\exists, \forall\}$
A_c
Asserted class hierarchy
Reasoner (R_1)
$(I/P)\,to\,R_1$
Make Required Changes
Consistant?
Yes
No
I_c
Inferred class hierarchy
Ontology $\alpha_{(Domain)}$
End

Fig. 2. Schematic diagram for representing ontology development process

4 Experimental Design and Result

This section of the paper selects an ontology development tool “Protégé–5.1.0” developed by “$StanfordUniversity$” for explaining the experimental design setup and resulting ontology.

4.1 Ontology and Knowledge-Base Module Representation

- Define classes (C) refer ($eq^n - 1$), define properties (P), sub properties, inverse properties refer ($eq^n - 2$), define class hierarchy (individuals (I)) refer ($eq^n - 3$), then define relations in between individuals using properties.

Following Example 1 discuss a case by showing the components of the ontology (α_{ELE}).

Example 1. Class ($\boldsymbol{C}$) contains: Motor, Components.

Properties ($\boldsymbol{P}$) contains: hasProperty (hasBearing, hasBrushes, hasRotor, hasSpeed, etc.) & isPropertyof is the inverse property of hasProperty. It contains following elements (isBearingof, isBrushesof, isRotorof, etc.).

Individual ($\boldsymbol{I}$) contains: DCMotor, InductionMotor, SynchronousMotor, Bearing, Brushes, Rotor, Speed, etc.

Following Fig. 3 represent the schematic diagram of classes, properties, and individuals.

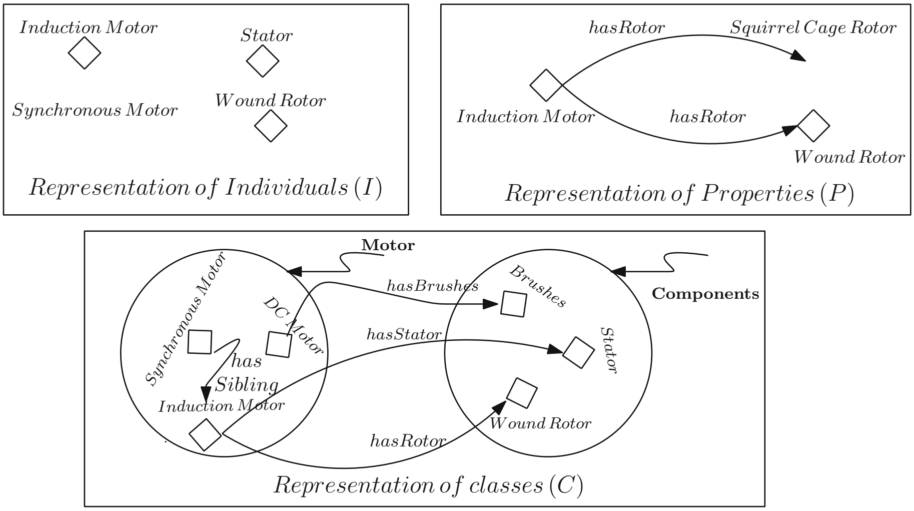


Fig. 3. Schematic diagram for representing classes, properties & individuals.

Following Fig. 4 shows the inferred class hierarchy I_c view of the ontology, that is the result of reasoner, for checking consistency and making assumptions about domain.

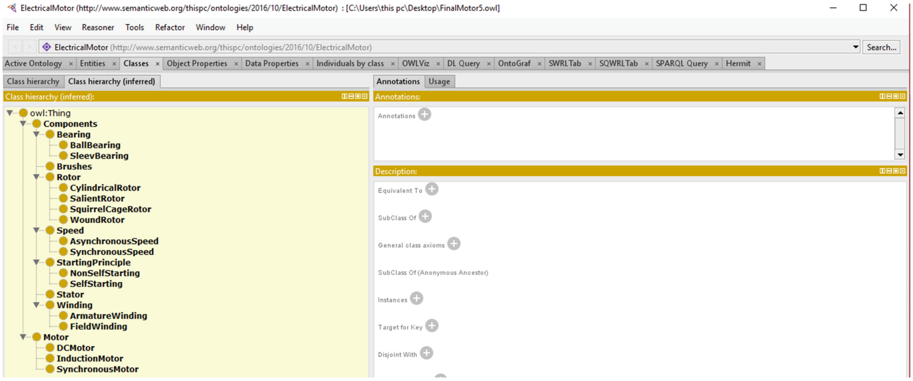

ElectricalMotor (http://www.semanticweb.org/thispc/ontologies/2016/10/ElectricalMotor) : [C:\Users\this pc\Desktop\FinalMotor5.owl]
File Edit View Reasoner Tools Refactor Window Help
ElectricalMotor (http://www.semanticweb.org/thispc/ontologies/2016/10/ElectricalMotor)
Search...
Active Ontology × Entities × Classes × Object Properties × Data Properties × Individuals by class × OWLViz × DL Query × OntoGraf × SWRLTab × SQWRLTab × SPARQL Query × Hermit ×
Class hierarchy
Class hierarchy (inferred)
Class hierarchy (inferred):
owl:Thing
Components
Bearing
BallBearing
SleevBearing
Brushes
Rotor
CylindricalRotor
SalientRotor
SquirrelCageRotor
WoundRotor
Speed
AsynchronousSpeed
SynchronousSpeed
StartingPrinciple
NonSelfStarting
SelfStarting
Stator
Winding
ArmatureWinding
FieldWinding
Motor
DCMotor
InductionMotor
SynchronousMotor
Annotations
Usage
Annotations:
Annotations
Description:
Equivalent To
SubClass Of
General class axioms
SubClass Of (Anonymous Ancestor)
Instances
Target for Key
Disjoint With

Fig. 4. Schematic diagram showing O/P (I_c) produced by the reasoner

4.2 Reasoner and Query Module Representation

After forming an ontology (α_{ELE}), one can request for information regarding the domain by using the query unit. There is a query unit "DL-Query" in "Protégé–5.1.0", which has the following format.

Query Format:

$$\begin{aligned}&\langle Property\,(p_i)\,,\;Restriction\,type\,(q_j)\,,Class\,Component\,(c_k)\rangle\\&\qquad\qquad\qquad\qquad:\{p_i \in Set\,[P]\,,q_j \in Set\,[Q]\,,c_k \in Set\,[C]\}\end{aligned} \tag{10}$$

The $Set\,[Q]$ (set of quantifiers) have following attributes:

$$\left\{\begin{array}{l}Some\,(existential)\,,Only\,(universal)\,,Min\,(min\,cardinality)\,,\\Exactly\,(exact\,cardinality)\,,Max\,(max\,cardinality)\end{array}\right\} \tag{11}$$

We select one of the property (p_i) from the property $Set\,[P]$, opt a restriction type (q_j) in among of available restriction types $Set\,[Q]$, then select a class component c_i in among of the available classes $Set\,[C]$. This triplet combination is used to find out the required information from the domain ontology.

4.3 Cognitive-Behaviour Modelling for Decision Making

To provide the cognitive decision-making capability to the system, one has to make following arrangements at the user interface module (UIM):

While answering the query one can include the cognitive behaviour (C_b), that has boolean values $[yes/no]$. This feedback mechanism can be further useful for the updation and amendment of new rules to the domain ontology.

Answer Format: It has three tuples $\langle Que, Ans, C_b\rangle$.

$$\langle Question\,(Que)\,,\;Answer\,(Ans)\,,\;CognitiveBehaviour\,(C_b)\rangle \tag{12}$$

Following Examples (2 and 3) explain (Eq. 10), by taking the components of ontology (α_{ELE}).

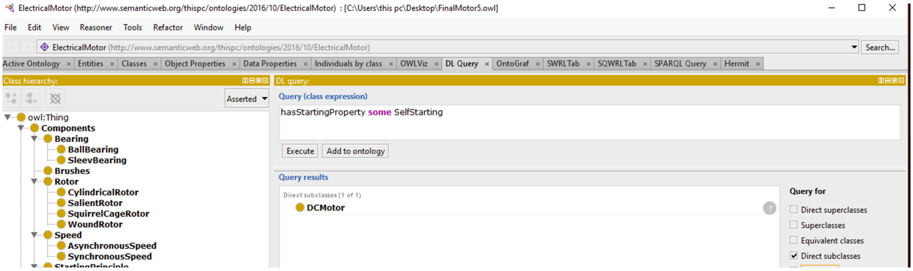
Example 2.

$$\left\{ \begin{array}{lcl} Que & : & \text{“?”} \;\; \text{“}hasStartingProperty\text{”} \;\; \text{“}some\text{”} \;\; \text{“}SelfStarting\text{”} \\ Ans & : & \text{“}DCMotor\text{”}. \\ C_b & : & \text{“}Satisfied\text{”} - (yes/no). \end{array} \right\}$$

Example 3.

$$\left\{ \begin{array}{lcl} Que & : & \text{“?”} \;\; \text{“}hasSpeed\text{”} \;\; \text{“}some\text{”} \;\; \text{“}AsynchronousSpeed\text{”} \\ Ans & : & \text{“}InductionMotor\text{”}. \\ C_b & : & \text{“}Satisfied\text{”} - (yes/no). \end{array} \right\}$$

Following Fig. 5 shows the result of DL-Query.



ElectricalMotor (http://www.semanticweb.org/thispc/ontologies/2016/10/ElectricalMotor) : [C:\Users\this pc\Desktop\FinalMotor5.owl]

ElectricalMotor (http://www.semanticweb.org/thispc/ontologies/2016/10/ElectricalMotor)

DL query:

Query (class expression)

hasStartingProperty some SelfStarting

Query results

Direct subclasses (1 of 1)

DCMotor

Query for

☐ Direct superclasses
☐ Superclasses
☐ Equivalent classes
☑ Direct subclasses

Fig. 5. A schematic diagram to illustrate the O/P, produced by the (DL-Query)

5 Conclusion and Future Work

Work done in this paper can be summarized as follows: This paper presents an ontology development process, that further provides domain information for cognitive decision making. Here the author has developed an ontology (α_{ELE}) that represents electrical motor concepts. This ontology will help to the novice students, who wants to learn the electrical motor concepts. Students can ask the questions related to electrical motors using query module. Query module uses the reasoner for inferencing and establishing a relationship that is further used for answering the queries. For establishing cognitive decision making capability and feedback mechanism, which is useful for augmentation and updating the rules, we have included cognitive behaviour concept (C_b) at the user interface module (UIM). It has boolean values [yes/no]. It takes the feedback from the user, that is further used for amendment and providing weights to the answers. This weight function will help in selecting the best answer in among of the available answers. There are several other approaches for knowledge modelling (like Semantic network, NLP (Natural language processing)).

Finally, to improve the decision-making process author suggests the use of NLP approach. Future work will be carried out in the area of NLP development for improved decision making.

Acknowledgments. I would like to thank Dr. P.S. Satsangi Sahab for his continuous inspirations and blessings and to Mr. Ashish Chandiok for his valuable guidance and support.

References

1. Lee, K., Lee, J., Kwan, M.-P.: Location-based service using ontology-based semantic queries: a study with a focus on indoor activities in a university context. Comput. Environ. Urban Syst. **62**, 41–52 (2017)
2. Liu, J., Liu, L., Xue, Y., Dong, J., Hu, Y., Hill, R., Guang, J., Li, C.: Grid workflow validation using ontology-based tacit knowledge: a case study for quantitative remote sensing applications. Comput. Geosci. **98**, 46–54 (2017)
3. Raies, K., Khemaja, M., Mejbri, Y.: Automatic extraction of smart game based learning design expertise: an approach based on learning ontology. Innovations in Smart Learning. LNET, pp. 161–170. Springer, Singapore (2017). doi:10.1007/978-981-10-2419-1_23
4. Liu, J., Zheng, B.-J., Luo, L.-M., Zhou, J.-S., Zhang, Y., Yu, Z.-T.: Ontology representation and mapping of common fuzzy knowledge. Neurocomputing **215**, 184–195 (2016)
5. Morente-Molinera, J.A., Wikström, R., Herrera-Viedma, E., Carlsson, C.: A linguistic mobile decision support system based on fuzzy ontology to facilitate knowledge mobilization. Dec. Support Syst. **81**, 66–75 (2016)
6. Zhang, Y.-F., Tian, Y., Zhou, T.-S., Araki, K., Li, J.-S.: Integrating HL7 RIM and ontology for unified knowledge and data representation in clinical decision support systems. Comput. Methods Programs Biomed. **123**, 94–108 (2016)
7. Chuprina, S., Alexandrov, V., Alexandrov, N.: Using ontology engineering methods to improve computer science and data science skills. Procedia Comput. Sci. **80**, 1780–1790 (2016)
8. Maffei, A., Daghini, L., Archenti, A., Lohse, N.: CONALI ontology. A framework for design and evaluation of constructively aligned courses in higher education: putting in focus the educational goal verbs. Procedia CIRP **50**, 765–772 (2016)
9. Chuprina, S., Postanogov, I., Nasraoui, O.: Ontology based data access methods to teach students to transform traditional information systems and simplify decision making process. Procedia Comput. Sci. **80**, 1801–1811 (2016)
10. Wang, H., Wang, S.: Application of ontology modularization to human-web interface design for knowledge sharing. Expert Syst. Appl. **46**, 122–128 (2016)
11. Tadjine, Z., Oubahssi, L., Piau-Toffolon, C., Iksal, S.: A process using ontology to automate the operationalization of pattern-based learning scenarios. In: Zvacek, S., Restivo, M.T., Uhomoibhi, J., Helfert, M. (eds.) CSEDU 2015. CCIS, vol. 583, pp. 444–461. Springer, Cham (2016). doi:10.1007/978-3-319-29585-5_26
12. Chun, S.A., Geller, J.: Developing a pedagogical cybersecurity ontology. In: Helfert, M., Holzinger, A., Belo, O., Francalanci, C. (eds.) DATA 2014. CCIS, vol. 178, pp. 117–135. Springer, Cham (2015). doi:10.1007/978-3-319-25936-9_8
13. Nuntawong, C., Namahoot, C.S., Brückner, M.: A semantic similarity assessment tool for computer science subjects using extended Wu & Palmer's algorithm and ontology. In: Kim, K. (ed.) Information Science and Applications. LNEE, vol. 339. Springer, Heidelberg (2015). doi:10.1007/978-3-662-46578-3_118
14. Lalingkar, A., Ramnathan, C., Ramani, S.: Ontology-based smart learning environment for teaching word problems in mathematics. J. Comput. Educ. **1**(4), 313–334 (2014). Springer, Heidelberg

15. Chi, Y.-L., Chen, T.-Y., Tsai, W.-T.: Creating individualized learning paths for self-regulated online learners: an ontology-driven approach. In: Rau, P.L.P. (ed.) CCD 2014. LNCS, vol. 8528, pp. 546–555. Springer, Cham (2014). doi:10.1007/978-3-319-07308-8_52
16. Altun, A., Kaya, G.: Development and evaluation of an ontology based navigation tool with learning objects for educational purposes. In: Huang, R., Kinshuk, Chen, N.-S. (eds.) The New Development of Technology Enhanced Learning: Concept, Research and Best Practices. Lecture Notes in Educational Technology, pp. 147–162. Springer, Heidelberg (2014)

Parametric Optimization Based on Bacterial Foraging Optimization

Daria Zaruba(✉), Dmitry Zaporozhets, and Elmar Kuliev

Southern Federal University, Rostov-on-Don, Russia
daria.zaruba@gmail.com, elpilasgsm@gmail.com,
elmar_2005@mail.ru

Abstract. Today parametric optimization problems are widely used in different science and technology domains. These problems are weather forecasting, calculation of electromotor parameters as well as VLSI design problems which belong to NP-problems and do not have deterministic algorithms to solve it. So, it is necessary to develop up-and-coming heuristic methods to obtain quazi-optimal solutions during polynomial time. The paper deals with parametric optimization of technical objects. From the mathematical point of view the parametric optimization problem reduces to the global constrained continuous optimization. The formulation of parametric optimization problem is made. To solve this problem it is developed a stochastic algorithm based on foraging behavior of E.coli bacteria. A bacterial colony is considered as multiagent system in which each agent operates in autonomy according with quite elementary rules. Colony behavior is based on self-organization to reach common goals by low-level interconnection. The colony does not have centralized control. Conjunction of simple agents creates a behavioral strategy without any global control. To analyze the developed algorithm there were carried out a set of experiments, which confirm theoretical estimations and calculate optimal values of algorithm parameters. Experimental results show perspective of this approach.

Keywords: Parametric optimization · Swarm intelligence · Bioinspired search · Bacterial colony

1 Introduction

Nowadays, in terms of parametric optimization problems probabilistic methods based on simulation of creatures' behavior in nature are most often used. An experience has shown that multi-agent methods are efficient tool for optimization in various scientific and technical fields. To design high-performed mechanisms for specific optimization problems it is necessary to developed promising methods inspired by natural systems. There are algorithms based on evolutionary computation, ants, bacterial and particle swarm optimization methods [1–5].

In the paper authors suggest a probabilistic algorithm on the basis of multi-agent intelligence conception that is simulation of E.coli bacteria movement. The key idea is that during the movement in environment the bacterium strives towards nutritious areas and avoids harmful ones. In semisolid useful environment bacteria can create stable

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_6

spatiotemporal structures. Under specific conditions E.coli can group and show intelligence cooperative behavior.

To estimate the algorithm's quality and compare obtained results with another swarm intelligence algorithms there were conducted a computational experiment. On the basis of these researches the authors can confirm that the time complexity of the developed algorithm has polynomial complexity.

2 Formulation of Parametric Optimization Problem

During optimal design a mathematical model of technical object is formalized description of quality criterion which execute function, requirements, etc. [6, 7].

The parametric optimization problem is solved by finding such input circuit parameters that output parameters have specified characteristic but circuit elements and way of its connection remain the same.

Let n are control parameters in the vector X = (x1, x2, …, xn). Let F(X) is an objective function (OF), XO is its definition area. The vector X defines coordinates of point within definition area XO. If elements in X have only discrete values, then XO is a discrete set and the optimization problem belongs to discrete programming field.

The aim of parametric optimization algorithms is define such control parameter vector that the specified objective function posses optimal value.

During the mathematical model development it is necessary to calculate object parameters affecting on an optimal criterion. Then, there are defined parametric, discrete and functional restrictions of the technical object [5, 6].

Parametric restrictions are represented as follows:

$$x_i' \leq x_i'' \tag{1}$$

Where x_i is i-th parameter of the object; $x_i^{'}$ and $x_i^{'}$ are minimum and maximum values of the i-th parameter correspondingly.

Discrete restrictions are written as:

$$x_j = \{x_{j1}, x_{j2}, \ldots, x_{jm}\} \tag{2}$$

Where x_j is j-th parameter of the object; x_{jk} is acceptable value of j-th parameter (k = 1,2,…,m). Such restrictions are imposed on parameters value in connection with its physical entity.

Functional restrictions are constraining conditions of its values and are represented as follows:

$$g_i(x) \leq 0; g_j(x) = 0; g_k(x) < 0. \tag{3}$$

Functional restrictions involve strength, rigidity and stability conditions which provide desirable values of technical characteristics [7–10].

3 Biological Foundations of Bacterial Colony Behavior

The E.coli bacterium (Fig. 1) is the most studied. It contains a cell plasma membrane, a cell wall and a capsule with cytoplasm or nucleoid. Bacterium size is about 1 micron in diameter and 2 microns in length, weight is about 1 pg. Under certain conditions the E.coli is increased in size and divided into two descendant bacteria. Obtaining necessary nutrition and maintaining appropriate temperature (about 37 °C), reproduction takes about 20 min, hence, a bacteria population growths exponentially [7–9].

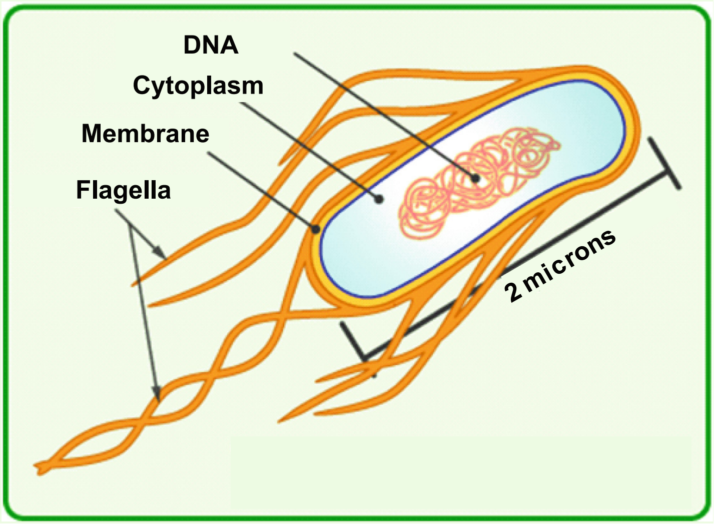


Fig. 1. The E.coli bacterium

Locomotion is achieved via a set of relatively rigid flagella that enable it to "swim" via each of them rotating in the same direction at about 100–200 revolutions per second. Each flagellum is a left-handed helix configured so that as the base of the flagellum (i.e., where it is connected to the cell) rotates counterclockwise, as viewed from the free end of the flagellum looking towards the cell, it produces a force against the bacterium so it pushes the cell. If a flagellum rotates clockwise, then it will pull at the cell. From an engineering perspective, the rotating shaft at the base of the flagellum is quite an interesting contraption that seems to use what biologists call a "universal joint" (so the rigid flagellum can "point" in different directions, relative to the cell). In addition, the mechanism that creates the rotational forces to spin the flagellum in either direction is described by biologists as being a biological "motor" (a relatively rare contraption in biology even though several types of bacteria use it). The motor is quite efficient in that it rotates a complete revolution using only about 1000 protons and thereby E.coli spends less than 1% of its energy budget for motility [10, 11].

An E.coli bacterium can move in two different ways: it can "run" (swim for a period of time) or it can "tumble," and it alternates between these two modes of operation its entire lifetime (i.e., it is rare that the flagella will stop rotating).

The motion patterns that the bacteria will generate in the presence of chemical attractants and repellents are called "chemotaxis." For E.coli, encounters with serine or aspartate result in attractant responses, while repellent responses result from the metal ions Ni and Co, changes in pH, amino acids like leucine, and organic acids like acetate. What is the resulting emergent pattern of behavior for a whole group of E.coli bacteria? Generally, as a group they will try to find food and avoid harmful phenomena, and when viewed under a microscope, you will get a sense that a type of intelligent behavior has emerged, since they will seem to intentionally move as a group.

4 Model of Bacterial Foraging Optimization Algorithm

In terms of search optimization problems a chemotaxis can be viewed as an heuristic optimization mechanism using all resources to find new areas with a great number of useful resources [12, 13].

A classical bacterial foraging optimization (BFO) algorithm is based on three mechanisms: chemotaxis, reproduction and dispersal.

Let $X_{i,r,l} - (|X| \times l)$ is a current position of a bacterium $s_i \in S$ at iteration t, step of reproduction r, step of elimination l. There are $i \in [1 : |S|]$, $t \in [1 : \hat{t}]$, $l \in \left[1 : \widehat{t^l}\right]$, where $|S|$ is an even number of agents in colony S, $\hat{t}$, $\widehat{t^r}$, $\widehat{t^l}$, are total number of iteration (steps of chemotaxis), steps of reproduction and elimination. Besides that, corresponding value of objective function is denoted as $\varphi_{i,r,l}$.

Chemotaxis. In terms of bacterial foraging optimization algorithm a local optimization is realized as chemotaxis procedure. The next position $X^{'}_{i,r,l}$ of the bacterium s_i is calculated as

$$X^{'}_{i,r,l} = X_{i,r,l} + \lambda_i \frac{V_i}{\|V_i\|_E} \tag{4}$$

Where V_i is a vector of chemotaxis step for the bacterium s_i; λ_i is a current value of step. During the swimming vector V_i remains constant at the next iteration, i.e. $V^{'}_i = V_i$. When bacterium is tumble, vector $V^{'}_i$ is a random vector in the interval [−1;1].

Reproduction. The main aim of the reproduction mechanism is increasing of convergence rate of the algorithm by means of search space narrowing. Let h_i is a "health" of bacterium s_i. Then, for all trajectory points the total value of objective function is calculated as

$$h_i = \sum\nolimits_{\tau=1}^{t} \varphi_{i,r,l}(\tau). \tag{5}$$

Further, it is denoted h_i, $i \in [1 : |S|]$ and bacteria are sorted in the decreasing order of its health. As a result we obtain a linear list.

Elimination. To find a global optimum of the objective function there are not enough chemotaxis and reproduction mechanisms since they are not solve a preliminary convergence problem. To overcome the weakness of the method it is used an elimination procedure.

This procedure initializes after reproduction and consists in the following. According with predetermined probability ξ_e it is selected n bacteria $s_{i_1}, s_{i_2}, \ldots, s_{i_n}$ in a random way and delete it from population. Instead of these in random point of search space there are generated new bacteria (agents) with the same number. Should be noted, after elimination procedure a number of agents in the population remains constant [14].

5 Description of the Algorithm

A block diagram of the developed algorithm for the parametric optimization problem is shown on Fig. 2. To illustrate key features of the BFO algorithm let us consider it in the context of a VLSI placement problem.

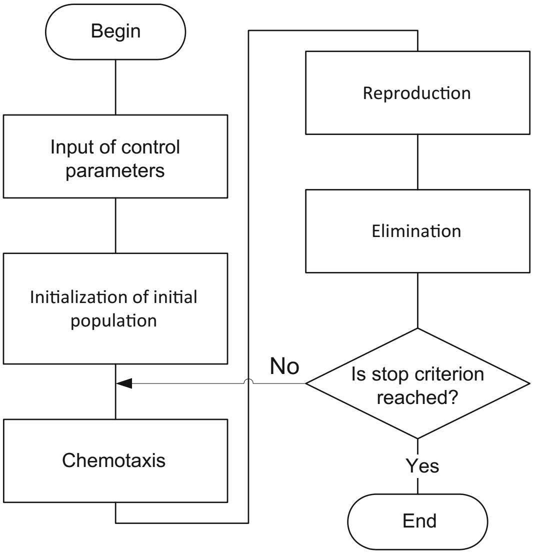

Begin
Input of control parameters
Initialization of initial population
Chemotaxis
Reproduction
Elimination
No
Is stop criterion reached?
Yes
End

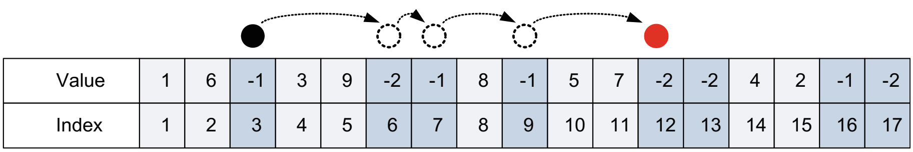
Fig. 2. The block diagram of the BFO algorithm

On the basis of the heuristic described above, the authors distinguish the following steps: input of control parameters, initialization of initial population, chemotaxis, reproduction, elimination.

The VLSI algorithm based on the BFO works with the following control parameters: a set of alternative solutions in the population N_S, a number of generations (iteration) T, initial value of "health" for each alternative solution H_0, a number of chemotaxis step N_x.

During initialization stage it is generated a set of alternative solutions for each of which OF values are calculated. A set of solutions can be obtained in previous stages, for example, as a result of a hierarchical algorithm or generates by the random way. Also, at this stage for each alternative solution it is necessary to define an initial index of gene (element of alternative solution) that began the agent's movement. The authors suggest to distinguish two direction: increasing and decreasing of the gene's index. In initialization stage an agent's direction is random.

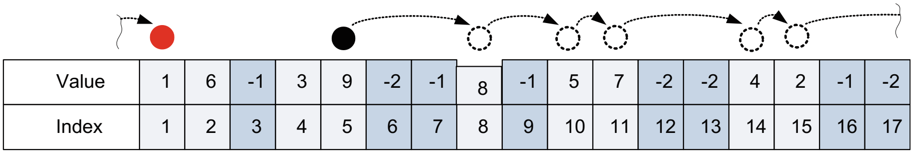
Also, the authors introduce the step of agent that means a number of genes the same type between initial and finite indexes. In terms of the developed algorithm there are used two types of genes: operands (encoded by positive integer) and operates (encoded by negative integer). So, a gene, situated in gene-operand at initial step, can move only in gene-operands, and a gene, situated in gene-operator at initial step, can move only in gene-operators. When the first of the last gene is reached, the calculation of finite gene's index is conducted repeatedly. The example of agent movement with the initial index 3 in gene-operators with step 4 is shown on Fig. 3.



Value	1	6	-1	3	9	-2	-1	8	-1	5	7	-2	-2	4	2	-1	-2
Index	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

Fig. 3. Agent movement in gene-operators

Example of agent movement with the initial index 5 in gene-operands with step 6 is shown on Fig. 4.



Value	1	6	-1	3	9	-2	-1	8	-1	5	7	-2	-2	4	2	-1	-2
Index	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

Fig. 4. Agent movement in gene-operands

A result of agent movement is a pair permutation of genes with initial and finite index. After that, an increment of the OF value before and after permutation are calculated. The value of increment is summed with a current value of agent's "health". Here, the "health" is total increment of the OF value during all steps of chemotaxis. Obviously, that increment can be positive and negative. If after regular step of chemotaxis the value of "health" is less or equal to 0, then movement is stopped for current alternative solution. Note, an agent changes its direction at the next step if increment will be negative.

When chemotaxis is stopped for each alternative solution, there is a reproduction is initialized. Reproduction contains the following steps:

- All sets of alternative solutions (populations) are sorted in decreasing of OF values.
- The first half of solutions is doubled. Alternative solutions from the first half of the sorted list are copied and migrate to the current population.

- For copies the initial index of agent is changed by a random way and the "health" value is equal to H_0.

Next, there is elimination for saving constant quantity of alternative solutions in the population N_S. Elimination contains the following steps:

- Population is sorted in increment of the value of "health".
- All alternative solutions with "health" less or equal to 0 are removed from the population.
- If current size of population $N_S^{cur} < N_S$, then there are generated additional quantity of solutions N_S - N_S^{cur}.
- If current size of population $N_S^{cur} > N_{S,}$ then alternative solutions with the lowest value of "health" are deleted until N_S^{cur} will be equal to N_S^{cur}.

The elimination stage is last step of iteration in terms of VLSI fragments placement on the basis of BFO. After that optimization continues iteratively until a stop criterion will be reached. Here, a stop criterion is a number of iteration in the algorithm.

6 Experiments

The authors developed software in the Borland C++ Builder™ 6.0. Testing of the developed algorithm was carried out on AMD FX(tm)-8121 Eight-Core Processor 3.10 GHz, RAM 4,00 Gb.

To conduct computational experiments there was developed software for VLSI fragments placement.

The obtained results allow to define a dependence of algorithm execution time on input parameters (Fig. 5).

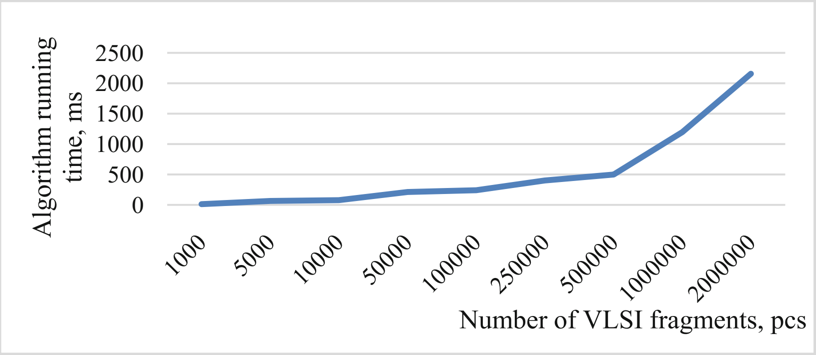


Fig. 5. Dependence of algorithm execution time on input parameters

The algorithm time complexity is represented as $O(n^2)$, where n is a number of input data.

Also, it was considered a dependence of algorithm execution time on a number of iteration (Fig. 6).

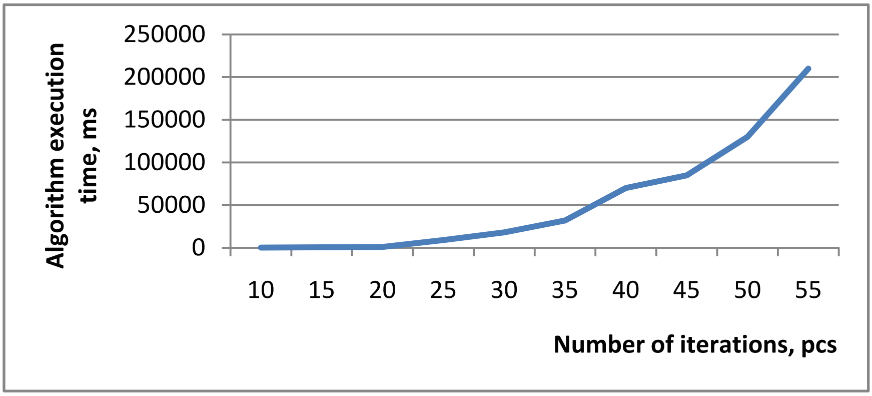


Fig. 6. Dependence of algorithm execution time on a number of iteration

The time complexity of this dependence is represented as O(n^4), where n is a number of iterations.

To estimate a quality of obtained solutions there were compared the BFO algorithm with well known VLSI fragment placement algorithms such as ant colony optimization and genetic algorithms (ACO and GA). The results of comparison are shown on Table 1 and Fig. 7.

Table 1. The results of comparison

Sr. No	A number of elements	Ser. size	ACO	GA	BFO
			Average value of OF, m	Average value of OF, m	Average value of OF, m
1	500	20	0,000125	0,000321	0,00013
2	1000	20	0,00412	0,005121	0,004112
3	2000	20	0,009124	0,014101	0,009141
4	3000	20	0,018121	0,021231	0,016121
5	5000	20	0,04101	0,051211	0,040012
6	8000	20	0,100231	0,105121	0,101643
7	10000	20	0,191231	0,2412	0,195512
8	15000	20	0,248	0,312	0,247
9	50000	20	1,36	1,98	1,34
10	100000	20	3,2	3,9	3,01
11	500000	20	18,6	19,6	18,33
12	750000	20	22,32	24,12	22,1
13	1000000	20	29,98	31,06	29,23

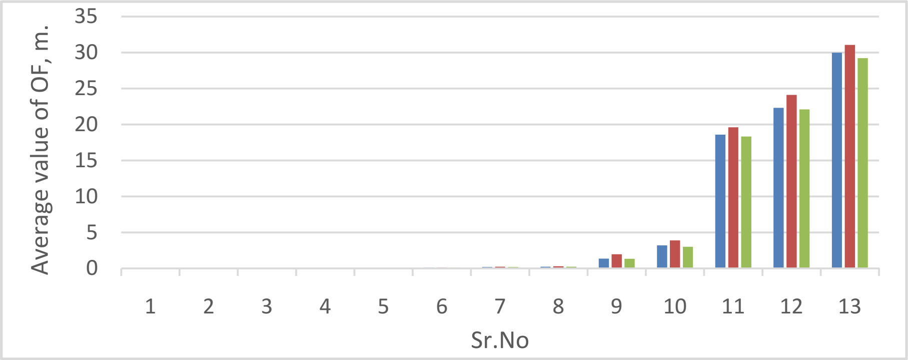


Fig. 7. Comparison of experimental results

7 Conclusion

In the paper the formulation of parametric optimization problem was made. To solve it the authors suggested the optimization method which simulates a behavior of bacterial colony. This algorithm allows to control a search process and eliminate the preliminary convergence problem. Software was developed in C++. To estimate time complexity of the developed algorithm there were carried out computational experiment. So, there were obtain empirical decencies, range of variation of input parameters, as well as give recommendations for its optimal choice. On the basis of obtain results the authors conclude that algorithm time complexity do not exceed polynomial complexity.

Acknowledgements. This research is supported by the Council for Grants (under RF President), the project #MK-92.2017.8.

References

1. Kureichik, V.V., Kureichik, V.M., Malioukov, S.P., Malioukov, A.S.: Algorithms for Applied CAD Problems. Springer, Berlin (2009). 487 p.
2. Alpert, C.J., Dinesh, P.M., Sachin, S.S.: Handbook of Algorithms for Physical design Automation. Auerbach Publications Taylor & Francis Group, Boca Raton (2009)
3. Zaruba, D., Zaporozhets, D., Kureichik, V.: Artificial bee colony algorithm—a novel tool for VLSI placement. In: Abraham, A., Kovalev, S., Tarassov, V., Snášel, V. (eds.) Proceedings of the First International Scientific Conference Intelligent Information Technologies for Industry (IITI 2016). AISC, vol. 450, pp. 433–442. Springer, Cham (2016). doi:10.1007/978-3-319-33609-1_39
4. Zaruba, D., Zaporozhets, D., Kureichik, V.: VLSI placement problem based on ant colony optimization algorithm. In: Silhavy, R., Senkerik, R., Oplatkova, Z., Silhavy, P., Prokopova, Z. (eds.) Artificial Intelligence Perspectives in Intelligent Systems. AISC, vol. 464, pp. 127–133. Springer, Cham (2016). doi:10.1007/978-3-319-33625-1_12

5. Kureichik, V., Kureichik, V., Zaruba, D.: Hybrid bioinspired search for schematic design. In: Abraham, A., Kovalev, S., Tarassov, V., Snášel, V. (eds.) Proceedings of the First International Scientific Conference Intelligent Information Technologies for Industry (IITI 2016). AISC, vol. 451, pp. 249–255. Springer, Cham (2016)
6. Kureichik, V., Kureichik, V., Bova, V.: Placement of VLSI fragments based on a multilayered approach. In: Silhavy, R., Senkerik, R., Oplatkova, Z., Silhavy, P., Prokopova, Z. (eds.) Artificial Intelligence Perspectives in Intelligent Systems. AISC, vol. 464, pp. 181–190. Springer, Cham (2016). doi:10.1007/978-3-319-33625-1_17
7. Kureichik, V.V., Kravchenko, Y.A.: Bioinspired algorithm applied to solve the travelling salesman problem. World Appl. Sci. J. **22**(12), 1789–1797 (2013)
8. Kar, A.K.: Bio inspired computing - a review of algorithms and scope of applications. Expert Syst. Appl. **59**, 20–32 (2016)
9. Lim, S.K.: Practical Problems in VLSI Physical Design Automation. Springer Science +Business Media B.V, Heidelberg (2008)
10. Yang, C., Ji, J., Liu, J., Yin, B.: Bacterial foraging optimization using novel chemotaxis and conjugation strategies. Inf. Sci. **363**, 72–95 (2016)
11. Zhao, W., Wang, L.: An effective bacterial foraging optimizer for global optimization. Inf. Sci. **329**, 719–735 (2016)
12. Kureichik, V.V., Zaruba, D.V.: The bioinspired algorithm of electronic computing equipment schemes elements placement. In: Silhavy, R., Senkerik, R., Oplatkova, Z., Silhavy, P., Prokopova, Z. (eds.) Artificial Intelligence Perspectives in Intelligent Systems. AISC, vol. 347, pp. 51–58. Springer, Cham (2015). doi:10.1007/978-3-319-18476-0_6
13. Zaporozhets, D., Zaruba, D.V., Kureichik, V.V.: Hierarchical approach for VLSI components placement. In: Silhavy, R., Senkerik, R., Oplatkova, Z., Silhavy, P., Prokopova, Z. (eds.) Artificial Intelligence Perspectives in Intelligent Systems. AISC, vol. 347, pp. 79–87. Springer, Cham (2015). doi:10.1007/978-3-319-18476-0_9
14. Hernández-Ocaña, B., Mezura-Montes, E., Pozos-Parra, Ma.D.P. Evolutionary bacterial foraging algorithm to solve constraint numerical optimization problems. In: CEUR Workshop Proceedings, vol. 1659, pp. 58–65 (2016)

Hybrid Approach for Graph Partitioning

Vladimir Kureichik, Daria Zaruba(✉), and Vladimir Kureichik Jr.

Southern Federal University, Rostov-on-Don, Russia
vkur@sfedu.ru, daria.zaruba@gmail.com,
kureichik@yandex.ru

Abstract. The paper discusses hybrid algorithm based on the elements grouping for the graph partitioning problem which is one of the most important optimization problem in the decision making. This problem belongs to the NP-class problem that is there are no precise methods to solve this problem. Also we formulate the partitioning problem and choose an optimization criterion. The developed hybrid algorithm obtains optimal and quasi-optimal solutions during polynomial time. The distinguish feature of this algorithm is grouping of elements with the same attributes. According to aggregation mechanism, definite variety of fractals can be obtained in the process of random growth. Aggregation is a random process and creation of clusters is based on minimal arrays in graph. To compare obtained results with known analogous algorithms we developed software which allows to carry out experiments. As a result theoretical estimations of algorithm efficiency were confirmed by experimental results. The time complexity of the hybrid algorithm cam be represented as $O(nlogn) - O(\alpha n^3)$.

Keywords: Graph theory · Partitioning problem · Elements grouping · Optimization · Decision making

1 Introduction

Today, the key problems in science and technology fields are a development of methods and models for effective decision making. Furthermore, the main problems here are hybrid algorithm development for decision support systems; simulation of evolutionary development principles; adaptation and interconnection with environment.

One of the most important optimization problems in decision making is a graph partitioning into predetermined or random number of parts [1–3]. The graph partitioning problem has a lot of practical applications. It is used for design of automated controls and computer engineering devices, management systems, computer and engineering networks and for different artificial intelligence problems. Note, the partitioning problem belongs to NP-complete class, i.e. there are no effective algorithms to solve it with polynomial time complexity. In this regards there is necessary to develop different heuristics based on hybrid approaches of natures simulating systems [3, 4].

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_7

2 Formulation of Graph Partitioning Problem

Let us formulate the graph partitioning problem. Given graph $G = (X, E, W)$, where X is a set of graph vertices, E is a set of edges, W is a total vertices weigh. Vertex weight is associated with integral estimation containing various design and engineering restrictions on the investigate model. Besides that, values of $w_i \in W$ do not exceed a threshold.

Let $B = \{B_1, B_2, \ldots, B_s\}$ is a set of parts $B_1, B_2, \ldots, B_s$ into graph G. Here, $B_1 \cap B_2 \cap \ldots \cap B_s = \emptyset$ and $B_1 \cup B_2 \cup \ldots \cup B_s = B$. Each part B_i contains elements $B_i = \{b_1, b_2, \ldots, b_n\}$, $n = |X|$. Then, the partition of a graph G into parts involves partition $B_i \in B$ which meet following requirements and restrictions:

$$(\forall B_i \in B)(B \neq \emptyset); \tag{1}$$

$$(\forall B_i, B_j \in B)\left(\left[B_i \neq B_j \to X_i \cap X_j \neq \emptyset\right] \wedge \left[\left(E_i \cap E_j = E_{\text{ij}}\right) \vee \left(E_i \cap E_j = \emptyset\right)\right]\right); \tag{2}$$

$$\cup_{i=1}^{s} B_i = B;\ \cup_{i=1}^{n} E_i = E;\ \cup_{i=1}^{n} X_i = X;\ \left|E_{ij}\right| = K_{ij}. \tag{3}$$

For graph G partitioning an objective function is written as

$$K = \frac{1}{2}\sum\nolimits_{i=1}^{n}\sum\nolimits_{j=1}^{n} K_{ij}, (i \neq j), \tag{4}$$

Where K_{ij} is a number of connections between B_i and B_j, l is a number of parts in partition, K is a total number of edges in partition.

The standard partitioning problem consists in minimization K($K \to min$). It allows to consider various criteria and design and technological restrictions of an artificial intelligent systems (AIS) model for decision making. The paper deals with a partitioning of graph G with minimization K and a restriction $W \leq \lambda(w_1 \leq \lambda_1, \ldots, w_s \leq \lambda_s)$, where $\lambda, \lambda_1, \ldots, \lambda_s$ are predetermined thresholds. Hence, a partitioning problem is that to find such partition B_1 from a set of potential partitions B in a graph or hypergraph G, that an objective function K is minimizing (or maximizing) and all restrictions (if its exist) are considered.

A graph partitioning belongs to discrete constrained optimization problems because of an objective function is discrete and there are a lot of restrictions on variables. Besides that there are a lot of methods for its solution, but effective algorithms are known only for special class of problems such as linear, quadratic and convex programming. Simplex and non-linear programming methods (penalty function method) are commonly used [5–7]. But in term of partitioning problem the majority of methods cannot be used due to its discretisation that makes difficult of finding optimum value. So, these problems belong to special class of optimization combinatorial problems on graphs. In this case total number of alternative solutions is equal to number of permutations from n vertices in the graph, i.e. $C_n = n!$, and taking into account restrictions on subsets, to number of combination of n vertices and m parts [8], i.e.:

$$C_n^m = \frac{n!}{m!(n-m)!}. \tag{5}$$

Based on above, a graph partitioning based on exhaustive search is challenging due to exponential complexity of search process.

3 Mathematical Model Analysis

In the optimal graph partitioning problem as an alternative solution is a concrete partition, which meets the predetermined requirements, that allow to interpret a partition problem solution as an evolutionary process with vertices redistribution $x_i \in X$ within graph G.

The authors suggest a hybrid approach based on elements grouping and genetic search.

To population description there are insert two type of variable attributes representing quantitative and qualitative differences between chromosomes. Qualitative attributes divide a set of chromosomes into clearly distinguishable groups. Quantitative attributes represent continuous variability of search process and define by number.

During interconnection chromosomes with environment it genotype causes externally observable quantitative attributes that involve a fitness $\mu(P_k)$ of the chromosome P_k to environment and its phenotype. If environment will be represented as optimal criterion K, for each chromosome, a fitness $\mu(P_k)$ is a numerical value of function K defining for acceptable solution $P_k \in D$. In general case a fitness $\mu(P_k) \geq 0$ can be written as

$$\mu(P_k) = \begin{cases} Q^2(\vec{x}) & \mathit{for\,maximization\,of}\ Q(\vec{x}); \\ \frac{1}{Q^2(\vec{x})+1} & \mathit{for\,minimization\,of}\ Q(\vec{x}). \end{cases} \tag{6}$$

From (6) it follows that the more numerical fitness value $\mu(P_k)$, the better chromosomes adapt to environment. Hence, the purpose of the chromosome evolution is a fitness increasing. A fitness $\mu(P_k)$ in partitioning problem is coincide with optimum criterion K – total number of external edges between parts of partitions.

Obviously, that in a population P^t there are several different form of variable attributes, which allow to conduct division of population into subpopulation $P_i^t \subset P^t, i = \overline{1, V}$, which involve chromosomes with the same or sufficiently close forms of qualitative or/and quantitative attributes [8].

So, in the optimal partitioning problem to differentiate chromosomes $P_k \in P^t$ with respect to quantitative attribute it can be chosen such condition as a local population $P_1^t \subset P^t$ contains only chromosomes in which value of K do not exceed a predetermined value. Then, another local population contains those chromosomes P_k which are not in P_1^t.

4 Elements Grouping Algorithm

To decrease the complex of the partitioning problem it is need to reduce it dimension by decomposition the problem on sub problem.

The suggested algorithm architecture is shown on the Fig. 1.

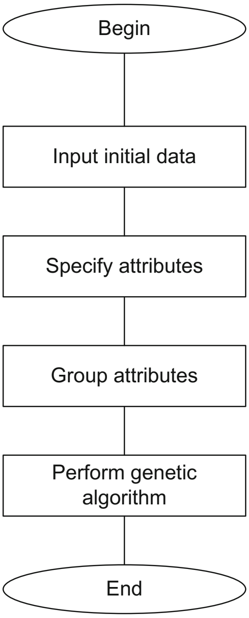


Fig. 1. Architecture of the partitioning algorithm

In terms of solutions the graph partitioning problem there are need to group elements with the same attributes [8–11]. For this purpose the creation of quazi-optimal clusters based on fractals aggregation is used [12]. According to aggregation mechanism, definite variety of fractals can be obtained in the process of random growth. Aggregation is a random process and creation of clusters is based on minimal arrays in graph [13].

For simulation of input data the authors suggest the following matrix:

$$R=\begin{array}{c||cccccccccccc||}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\
A_1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\
A_2 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\
A_3 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\
A_4 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\
A_5 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
A_6 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\
A_7 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \\
A_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}.$$

Here, matrix columns correspond to graph elements, and matrix rows - attributes or elements functions. For example, a digit at the intersection of the row A_4 and the column 4 means that the element 4 has the attribute A_4.

There is need to group elements with maximum number of the same attributes. This problem belongs to partitioning problems and for its solution it is suggested a hybrid approach.

The main idea of a hybrid approach is as follows. At first, a matrix model is analyzed. As a result, there are not considered rows of matrix with ones and zeros or containing about 1%. These attributes does not affect on grouping because all or almost all elements will be included in groups. Then, in the considered example rows A_5 and A_8. Next, the matrix R is sorted. In this case, a row with the maximum number of ones is at the first place. So, the matrix R can be written as:

$$\begin{array}{r||cccccccccccccccc||}
 & 12 & 4 & 7 & | & 8 & 9 & 11 & & 1 & 5 & 6 & | & 2 & 3 & 10 \\
A_1 & 1 & 0 & 1 & | & 1 & 1 & 1 & & 1 & 1 & 0 & | & 0 & 0 & 0 \\
A_2 & 1 & 1 & 1 & | & 1 & 0 & 1 & & 0 & 0 & 0 & | & 0 & 1 & 0 \\
R_1 = A_3 & 1 & 0 & 1 & | & 0 & 1 & 1 & & 1 & 1 & 0 & | & 0 & 1 & 0 \\
A_4 & 1 & 1 & 0 & | & 1 & 0 & 0 & & 0 & 0 & 1 & | & 1 & 0 & 1 \\
A_6 & 0 & 1 & 0 & | & 1 & 1 & 0 & & 0 & 1 & 1 & | & 0 & 0 & 1 \\
A_7 & 1 & 1 & 1 & | & 0 & 1 & 1 & & 1 & 0 & 1 & | & 1 & 0 & 0
\end{array}$$

Then, the matrix R is divided into two parts: $R_1 = \{12, 4, 7, 8, 9, 11\}$ with "promising" chromosomes and $R_2 = \{1, 5, 6, 2, 3, 10\}$ with "unpromising" chromosomes.

To parallelize search process R_1 and R_2 can be divided into $R_1^{'}, R_1^{''}$ and $R_2^{'}, R_2^{''}$. This process can be continued until partition to genetic algorithm processing will be obtained. It is assumed element-by-element addition of rows in matrix R according to following rules: 1 + 1 = 1; 1 + 0 = 0; 0 + 0 = 0 [10]. For example, according to addition for "promising" columns of the matrix R in $R_1^{'}, R_1^{''}$ (12 + 4; 12 + 7; 4 + 7) we obtain:

$$+\begin{array}{r} 12:1\;1\;1\;1\;0\;1 \\ \underline{4:0\;1\;0\;1\;1\;1} \\ (12,4):0\;1\;0\;1\;0\;1 \end{array} \qquad +\begin{array}{r} 12:1\;1\;1\;1\;0\;1 \\ \underline{7:1\;1\;1\;0\;0\;1} \\ (12,7):1\;1\;1\;0\;0\;1 \end{array} \qquad +\begin{array}{r} 4:0\;1\;0\;1\;1\;1 \\ \underline{7:1\;1\;1\;0\;0\;1} \\ (4,7):0\;1\;0\;0\;0\;1 \end{array}$$

The group (12, 4) has three common attributes, the group (12, 7) has four common attributes and the group (4, 7) has two common attributes. According to the fractal aggregation we obtain:

$$+\begin{array}{r} 8:1\;1\;0\;1\;1\;0 \\ \underline{9:1\;0\;1\;0\;1\;1} \\ (8,9):1\;0\;0\;0\;1\;0 \end{array} \qquad +\begin{array}{r} 8:1\;1\;0\;1\;1\;0 \\ \underline{11:1\;1\;1\;0\;0\;1} \\ (8,11):1\;1\;0\;0\;0\;0 \end{array} \qquad +\begin{array}{r} 9:1\;0\;1\;0\;1\;1 \\ \underline{11:1\;1\;1\;0\;0\;1} \\ (9,11):1\;0\;1\;0\;0\;1 \end{array}$$

The group (8, 9) has two common attributes, the group (8, 11) has two common attributes and the group (9, 11) has three common attributes. Only groups with three or more common attributes are fixed. In a similar way in “promising” parts $R_2^{'}, R_2^{''}$ we obtain following groups: (1, 5) – 2 common attributes; (1, 6) – one; (5, 6) – one; (2, 3) – zero; (2, 10) – one; (3, 10) – zero. Obtained solutions are sorted and a population P is formed: $P = \{(12,7),(12,4),(9,11),(4,7),(8,9),(8,11),(1,5),(2,10)\}$. For genetic algorithm let perform a crossover operator;

<table>
<tr><td>P$_1$: 12 | 7</td><td>P$_2$: 12 | 4</td><td>P$_5$: 8 | 9</td><td>P$_6$: 8 | 11</td></tr>
<tr><td>P$_3$: 9 | 11</td><td>P$_4$: 4 | 7</td><td>P$_7$: 1 | 5</td><td>P$_8$: 2 | 10</td></tr>
<tr><td>P’$_1$: 12 | 11</td><td>P’$_2$: 12 | 7</td><td>P’$_5$: 8 | 5</td><td>P’$_6$: 8 | 10</td></tr>
<tr><td>P’$_3$: 9 | 7</td><td>P’$_4$: 4 | 4</td><td>P’$_7$: 1 | 9</td><td>P’$_8$: 2 | 11</td></tr>
</table>

After standard single-point crossover operator the group (12, 11) has four common attributes, (9, 7) – three; $P_2^{'}$ coincide with P_1 and does not considered; $P_4^{'}$ is illegal group; the group (8, 5) has two common attributes, (8, 5) – two, (1,9) – three, (8, 10) – two, (2, 11) – one. Note, a greedy crossover operator allows to avoid illegal groups. Now, we obtain a new population $P_1 = \{(12,7),(12,11),(9,11),(12,4),$ $(9,7),(1,9),(8,5),(8,10)\}$.

Here, size of population P_1 is equal to P. But, size of population can be varying according with the quality of obtained solutions. Let unite all groups according with maximum common attributes. Then, obtain following groups: (12, 7, 11) – four common attributes; (12, 7, 9) – three, (12, 11, 9) – three, (7, 11, 9) – thee. As a result we obtain group (12, 7, 11, 9, 1) which has three common attributes.

Time complexity of this algorithm for one generation can be represented as O(n)-O (nlogn). The algorithm can be performed in parallel way.

Approximate algorithm complexity can be written as

$$T \approx \left(N_{P}t_{p} + N_{P}t_{CO} + N_{P}t_{unity}\right)N_{G}, \tag{7}$$

where t_{CO} is a crossover operator complexity, t_{unity} is a unity operation complexity.

5 Experiments

In terms of this work there were developed a software environment on the basis of which the authors carried out a computational experiment. As a result theoretical estimations of algorithm efficiency were confirmed by experimental results. To calculate time complexity of the partitioning algorithm there were generated test graphs with specified characteristics. Table 1 and Fig. 2 show results of time complexity estimations.

Table 1. Results of computational experiments

Ser. no	Number of graph vertices, pcs	Number of connections, as % of max	Size of series, pcs	Execution time, sec
1	200	10–80	20	1, 35
2	400	10–80	20	2, 24
3	600	10–80	20	3, 66
4	800	10–80	20	7, 20
5	1000	10–80	20	13, 98
6	1200	10–80	20	22, 98
7	1400	10–80	20	40, 32
8	1600	10–80	20	65, 98
9	1800	10–80	20	84, 12
10	2000	10–80	20	103, 65
11	2200	10–80	20	116, 58
12	2400	10–80	20	130, 68
13	2600	10–80	20	151, 65
14	2800	10–80	20	168, 32
15	3000	10–80	20	180, 32
16	3200	10–80	20	199, 32
17	3400	10–80	20	215, 65
18	3600	10–80	20	240, 00
19	3800	10–80	20	269, 20
20	4000	10–80	20	310, 32

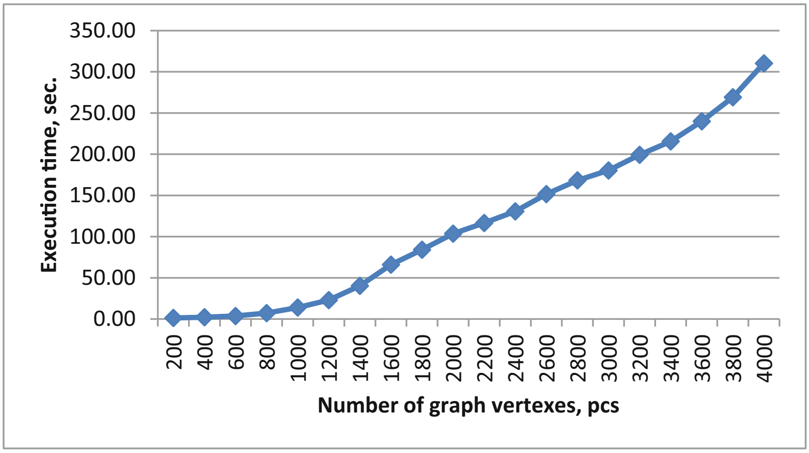


Fig. 2. Algorithm execution time

To estimate the efficiency of the suggested hybrid approach it was compared with simulated annealing and genetic algorithms. Table 2 and Figs. 3 and 4 show the comparison results [14].

Table 2. Comparison with analogous algorithms

Test graphs		Simulated annealing		Genetic algorithm		Hybrid algorithm	
Name	Number of elements, pcs	Execution time, sec	Number of interconnections, pcs	Execution time, sec	Number of interconnections, pcs	Execution time, sec	Number of interconnections, pcs
grph01	12506	813	654	973	545	1001	532
grph02	19342	942	1032	1065	879	1106	856
grph03	22853	965	2643	1165	2001	1178	1878
grph04	27220	932	3012	1189	2354	1201	2132
grph05	28146	989	2369	1210	1878	1215	1698
grph06	32332	1103	3365	1235	2310	1286	2145
grph07	45639	1265	3988	1277	2568	1301	2254
grph08	51023	1563	4578	1310	3654	1312	3214
grph09	53110	1565	4522	1389	3456	1386	3189
grph10	68685	1788	6598	1561	6100	1487	5789
grph11	70152	1799	6455	1620	6321	1532	5876
grph12	70439	1790	6788	1701	6512	1521	5963
grph13	83709	1806	7989	1754	6879	1598	6032

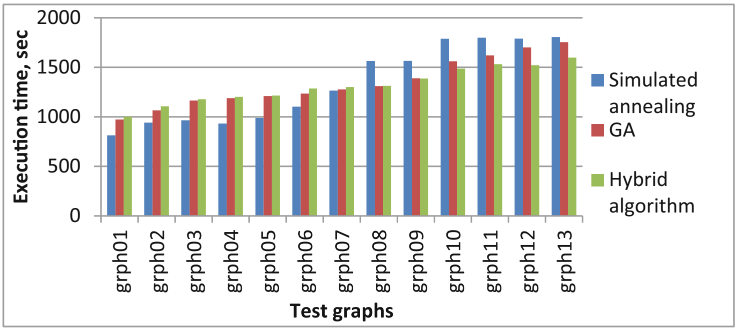


Fig. 3. Comparison of execution time

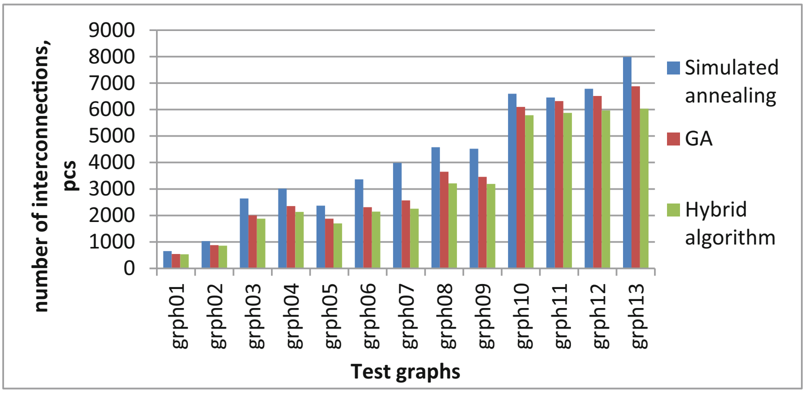


Fig. 4. Comparison of partitions quality

6 Conclusion

Partitioning algorithm realization shows advantages of the hybrid algorithm with non-standard search architecture in comparison with paired exchange and random algorithm. Genetic search management allow to find optimal parameters for the partition problem and modified genetic operators increase a partition quality.

The execution time of the hybrid algorithm is rising with the increase of a number of generations in the GA, but it is not significant and can be compensate by obtaining of a set of local optimum solutions. The genetic algorithm requires huge time consumption in comparison with paired exchange and random algorithm for graph partition. As for the speed of solution the hybrid algorithm almost coincide with iterative and sequential algorithms. Besides that the hybrid algorithm is much faster then simulated annealing and genetic algorithms. As distinguished from considered partition algorithms the hybrid algorithm allows to obtain quazi-optimal solutions in polynomial time. The time complexity of the hybrid algorithm cam be represented as $O(nlogn) - O(\alpha n^3)$.

Acknowledgements. This research is supported by internal grants of the Southern Federal University, the project #2.6432.2017.

References

1. Alpert, C.J., Dinesh, P.M., Sachin, S.S.: Handbook of Algorithms for Physical Design Automation. Auerbach Publications Taylor & Francis Group, Boca Raton (2009)
2. Zeng, J., Yu, H.: A study of graph partitioning schemes for parallel graph community detection. Parallel Comput. **58**, 131–139 (2016)
3. Kureichik, V.V., Zaruba, D.V.: Partitioning of ECE schemes components based on modified graph coloring algorithm. In: 12th IEEE East-West Design and Test Symposium (EWDTS 2014) (2014)
4. Kureichik, V.V., Zaporozhets, D.Y., Zaruba, D.V.: Partitioning of VLSI fragments based on the model of glowworm's behavior. In: Proceedings of the 19th International Conference on Soft Computing and Measurements (SCM 2016), art. no. 7519750, pp. 268–272 (2016)
5. Zaporozhets, D.Y., Zaruba, D.V., Kureichik, V.V.: Hybrid bionic algorithms for solving problems of parametric optimization. World Appl. Sci. J. **23**, 1032–1036 (2013)
6. Kureichik, V., Kureichik, V., Zaruba, D.V.: Combined approach to place electronic computing equipment circuit elements. In: Proceedings of 2015 IEEE East-West Design and Test Symposium (EWDTS 2015), art. no. 7493134 (2016)
7. Kureichik, V., Kureichik, V., Zaruba, D.: Hybrid bioinspired search for schematic design. In: Abraham, A., Kovalev, S., Tarassov, V., Snášel, V. (eds.) Proceedings of the First International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'16): Volume 2. AISC, vol. 451, pp. 249–255. Springer, Cham (2016). doi:10.1007/978-3-319-33816-3_25
8. Delgado, P., Desormeaux, W.J., Haynes, T.W.: Partitioning the vertices of a graph into two total dominating sets. Quaestiones Mathematicae **39**(7), 863–873 (2016)

9. Buluç, A., Meyerhenke, H., Safro, I., Sanders, P., Schulz, C.: Recent advances in graph partitioning. In: Kliemann, L., Sanders, P. (eds.) Algorithm Engineering. LNCS, vol. 9220, pp. 117–158. Springer, Cham (2016). doi:10.1007/978-3-319-49487-6_4
10. Sherwani, N.A.: Algorithms for VLSI Physical Design Automation, 3rd edn. Kluwer Academic Publisher, USA (2013)
11. Kacprzyk, J., Kureichik, V.M., Malioukov, S.P., Kureichik, V.V., Malioukov, A.S.: Experimental Investigation of Algorithms Developed. Studies in Computational Intelligence, pp. 211–223. Springer, Heidelberg (2009)
12. Bruglieri, M., Cordone, R.: Partitioning a graph into minimum gap components. Electron. Notes Discr. Math. **55**, 33–36 (2016)
13. Ariffin, W.N.M., Salleh, S.: Bi-partition approach of directed cyclic task graph onto multicolumn processors for total completion time minimization task assignment problem. In: AIP Conference Proceedings, vol. 1775, art. no. 030072 (2016)
14. http://vlsicad.ucsd.edu/UCLAWeb/cheese/ispd98.html

The Combined Method of Semantic Similarity Estimation of Problem Oriented Knowledge on the Basis of Evolutionary Procedures

V.V. Bova, E.V. Nuzhnov, and V.V. Kureichik(✉)

Southern Federal University, Rostov-on-Don, Russia
vvbova@yandex.ru, {nev,vkur}@sfedu.ru

Abstract. In the article authors proposed the method of problem-oriented knowledge elements search and similarity estimation in subject area ontology given in a form of semantic net. The knowledge relevance is estimated by closeness to a certain similarity estimation measure between concepts included in integrated ontology elements meta-descriptions of intellectual information systems interdisciplinary software environment. To calculate knowledge elements semantic closeness and coherence authors developed a combined model of semantic similarity estimation involving a set of interpreted measure of taxonomical and associative dependences represented in meta-descriptions. The methodology is based on relative position of ontology graph concepts in common hierarchy and on measures of similarity between properties in high-dimensional attribute space. Authors developed an algorithm to calculate parameters values of semantic similarity estimation model on the basis of evolutionary procedures and genetic optimum search. The proposed algorithm is based on the usage of evolutionary processes of reproduction, crossover, mutation and natural selection analogues. To analyze the developed method a set of experiments was carried out. The obtained data shows theoretical significance and prospects of such method and allows us to determine optimal values of algorithm parameters.

Keywords: Semantic similarity · Ontology · Semantic network · Semantic search · Semantic metadata · Genetic algorithms · Genetic operators

1 Introduction

Due to constant growth of information flows in different interdisciplinary technical, economical and social intelligent information systems (IIS), the development of new ways of distributed sources information representation, formalization, systematization, integration and search are relevant today [1–4]. One of the main functions of modern IIS involves semantic search of problem-oriented knowledge elements of a distributed and, thus, heterogeneous representation character.

In this paper the term ‘semantic search’ is considered as information search which provides comparison and similarity estimation of information objects on semantic level i.e. with the use of knowledge. Existing mechanisms of semantic search [5–7] are based on methods and approaches of knowledge subject area ontological conceptualization.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_8

This paper deals with the method of knowledge bases search, where metadata are formed on the basis of corresponding subject area ontologies represented in the form of semantic net. The knowledge relevance is estimated by closeness to a certain evaluation metric of similarity between concepts included in ontology elements semantic meta-description.

To calculate measures of semantic closeness and coherence between problem-oriented knowledge elements authors propose a combined model of semantic similarity estimation involving a set of interpreted taxonomical and associative meta-descriptions dependences of knowledge elements represented in ontology [8–10].

The algorithm of semantic similarity estimation is based on evolutionary procedures and genetic optimum search operators, which allows us to exclude uninformative or insignificant knowledge elements descriptions, and to manage speed of learning with the use of similarity threshold value assignment [11–13].

2 Problem Statement and Subject Area Analysis

The absence of a 'gold standard' for semantic similarity measures is a well-known problem. Many researchers are focused on the development of semantic similarity estimation and comparison methods used for a wide range of information search problems [2, 7–10].

In this paper factors defining selection of semantic similarity measures applied for formal representation (profile) of the user's query are proposed as follows:

- to select criteria composing the similarity measure: taxonomical relations between concepts – characteristics of ontological structures (the path length, hierarchy depth, etc.) and associative (horizontal) relations defining asymmetrical semantic similarity measure.
- to select criteria importance degree – importance coefficients in hybrid measure of computational semantics [14].

With the use of ontology in works [2, 10] authors proposed the method of special metadata type creation – meta-descriptions including sets of simple proposition statements of the form of 'subject *(s)*–predicate *(p)*–object *(o)*' which are referred as triplets *(t)* and represent main semantics of described knowledge elements. It is noted that such meta-descriptions are important sources of information for search implementation. With the use of meta-descriptions it is possible to significantly improve the search mechanisms functionality. Similarity estimation is called semantic similarity estimation if and only if it is determined on the basis of meta descriptions and query semantics [9].

Thus, to determine a degree of semantic similarity between knowledge elements it is proposed to introduce the measure of distance between their meta-descriptions. The measure represents the combination of several measures of distance between two vertexes (concepts or attributes) of shortest weighted path between ontology graph vertexes [15–17].

It is suggested that all concepts required to compare are located in the united ontology and, thus, in the united taxonomy [15]. If ontologies are separated, they should be united before the analysis [3].

In this paper problem statement authors use following descriptions of ontology components: the ontology O represents the sign system $O = <C, E, R, T>$, where C denotes a set of concepts (knowledge elements); E denotes a set of concepts examples; T denotes a set of predicates – relation types; R denotes a set of relations assigning following relation types between entities: taxonomical, attributive, quantitative, logical, etc.

Let us introduce following rules and constraints:

(1) On the basis of subject area ontology O, semantic meta-descriptions $m(c_i) = \{t_1, t_2, \ldots, t_{n(i)}\}$ are created for each knowledge elements $C = \{ci\}$, where $n(i)$ is a number of triplets in logical representation of a concept c_i; t_i denotes RDF-triplets having a form of tuples $<s_i, p_i, o_i>$, where s_i and o_i are included in the union of C_i and E_i, and p_i is include in R.

(2) Each query q created by the user from the set of queries Q consists of the set of triplets $q = \{t_1, t_2, \ldots, t_{n(q)}\}$, where $n(q)$ denotes a number of triplets included in the query q.

The assigned problem involves finding a weight function w, which determines the importance of any triplet $t \in T$ (where T denotes a set of possible triplets) when describing knowledge elements c_i from the query q: $0 \leq w(t, c_i) \leq 1$, where где $t \in T$, $c_i \in C$, $0 \leq w(t, q) \leq 1$, where $t \in T$, $q \in Q$.

For each query q it is required to determine a subset RES of the set of knowledge elements C, which includes relevant concepts for the assigned query q – the result set. C_i is considered as relevant to the query q, if and only if the semantic similarity estimation between them exceeds a certain threshold value of semantic closeness. Therewith, to estimate the similarity between knowledge elements and the query authors propose to use their semantic meta-descriptions [10].

3 The Combined Method of Semantic Similarity Estimation

The key moment in semantic search problem-solving includes the development of semantic similarity quantitative estimations. Existing methods of computational semantics can be subdivided in several categories: measures based on hierarchical structures – methods of conceptual taxonomical closeness estimation using different metrics of finding the length of the shortest path between subject area ontology graph vertexes [2, 16–18]; measures using non-hierarchical relations – methods of relational closeness estimation [5–7]; measures using attribute values [8–10].

The main problem of most measures based on ontological structures is symmetry. Expert analysis shows that similarity measure is not always symmetrical for both hierarchical and attributive relations [5, 6, 8–10]. The relevant problem is semantic similarity estimation between ontological elements that are not related hierarchically, but have concrete problem-specific (horizontal or associative) relation.

Thus, the most promising measures today are hybrid measures, which combine several methods considering ontology structures and relation semantics. This allows us to calculate semantic similarity estimations between ontology elements (concepts, examples, relations – predicates). Similarity estimations are referred as elementary estimations, and similarities between triplets are determined on their basis [2, 5]. Further, similarities estimations between triplets are used to determine similarity between meta-descriptions.

To determine semantic similarity between triplets of queries meta-descriptions M_q and triplets of concepts set M_c let us introduce metrics of distance between ontology nodes on the basis of taxonomy and concepts characteristics, and metrics of density and information value of concepts related thematically. Then, the modified similarity measure can be represented as follows:

$$SIM\left(M_q, M_c\right) = \sum\nolimits_{i=1}^{n} w(t,q)_i Sim^i(c_1, c_i), \tag{1}$$

Where Sim^i is a similarity measure based on a certain criterion, weight $w(t,q)_i$ determines the relative importance of query triplets criterion, weight summary equals to 1, n denotes a number of criteria.

To calculate Sim^i let us introduce the modification of asymmetrical similarity measure [5] considering all types of semantic relations R appropriate for triplet components similarity estimation. In suggested modification graph edges are assigned with a certain weigh coefficients depending on passing direction. This is based on the assumption that a child is more similar to a parent rather than opposite way.

1. For the relation 'parent-child' (*is-a*) two coefficients g and s are assigned, which represents similarity in direction of generalization and detailing.
2. For the relation *instanceOf* (connects concepts and concepts examples) two parameters $\delta, \gamma \in [0,1]$ are assigned, which represent similarity between the example and the concept and between the concept and the example.
3. Similarity coefficients assigned for the relation *sameAs* (synonyms) and *invertOf* (antonyms) equals to 1 and -1 respectively.
4. For other semantic relations r_i we assign weight coefficient ω, which represents semantic similarity in accordance with these relations.

Let us consider $D = \{c_1, \ldots, c_n\}$ as the path between entities c_1 and c_n (which can be concepts, examples or predicates). The path D has following characteristics:

1. *s*(*D*) is a number of edges in detailing direction;
2. *g*(*D*) is a number of edges in generalization direction;
3. *ic*(*D*) is a number of edges from the example to the concept;
4. *ci*(*D*) is a number of edges from the concept to the example;
5. *inv(D)* is a number of inverse relation edges;
6. *oth*(*D*) is a number of other relation edges.

The estimation of similarity between entities c_1 and c_2 in terms of criterion i and the path D is determined by the following formula:

$$Sim^i(c_1, c_2) = \max\nolimits_{j=1,..,m}\left\{\left(\left|(-1)^{inv(d_j)} s^{s(d_j)} * g^{g(d_j)} * \delta^{ic(d_j)} * \gamma^{ci(d_j)} * \omega^{oth(d_j)}\right|\right)\right\},) \tag{2}$$

where $d^1, \ldots, d^m$ denotes paths between vertexes c_1 and c_2.

To determine the density and information value of thematically related elements and their meta descriptions let us define the concept weight on the basis of occurrence degree. It is considered that the query concept weight depends on a number of meta descriptions concepts related with it *m(c_i)* which is represented by triplets $m(c_i) = \{t_1, t_2, \ldots, t_{n(i)}\}$, where $n(i)$ is a number of triplets in concept logical representation c_i [10].

$$w(t, q) = 1 + \ln\left(\varphi_{t,c_i}\left(1 + \sum\nolimits_{\mathrm{c_i \in C}} \varphi_{\mathrm{t,c_i}} SIM(c_1, c_i)\right)\right), \qquad (3)$$

Where $\varphi_{\mathrm{t,c_i}}$ is a coefficient of occurrence degree of query triplet q in meta description $m(c_i)$, the coefficient is assigned in the algorithm, $SIM(c_1, c_i)$ is a measure of semantic similarity between meta descriptions of concept vertexes in C.

4 Genetic Algorithm of Semantic Similarity Estimation

To improve the effectiveness of semantic similarity estimation and to determine semantically prioritized knowledge objects for the purpose of their representation in search model authors propose to use genetic algorithm (GA) which allows us to find suboptimal solutions in polynomial time effectively [19, 20]. GA is an heuristic search algorithm used for optimization and modeling problem solving by means of random selection, combination and variation of searched parameters with the use of mechanisms analogous to natural selection [20]. The generalized structure of genetic search is shown on the Fig. 1.

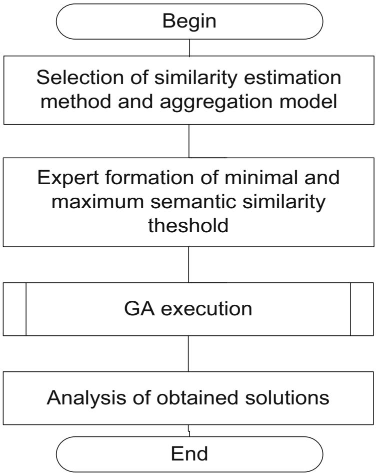

Begin
Selection of similarity estimation method and aggregation model
Expert formation of minimal and maximum semantic similarity theshold
GA execution
Analysis of obtained solutions
End

Fig. 1. The generalized architecture of genetic search

To determine optimal coefficients values authors defined the GA objective function with the use of similarity estimation maximization method:

$$\mathrm{F} = \max\left(SIM\left(M_q, M_c\right)\right). \tag{4}$$

The GA of model parameters values calculation for the purpose of semantic similarity estimation is shown on the Fig. 2.

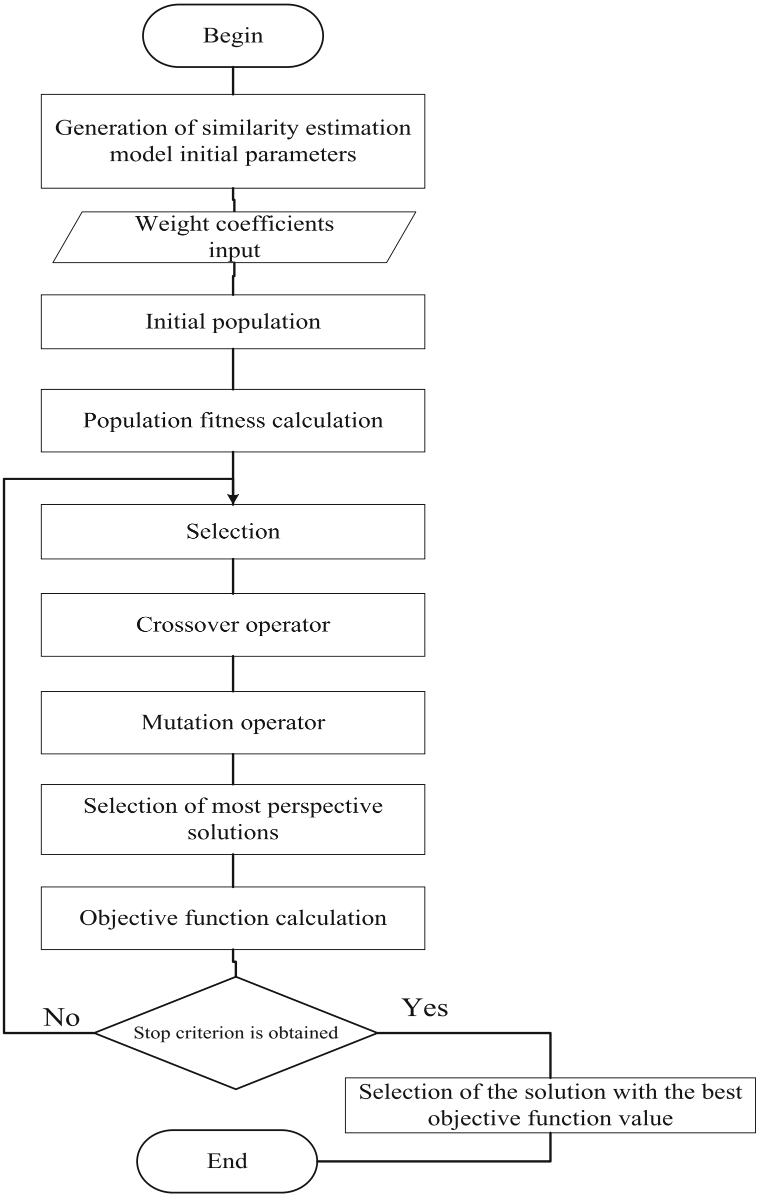

Begin
Generation of similarity estimation model initial parameters
Weight coefficients input
Initial population
Population fitness calculation
Selection
Crossover operator
Mutation operator
Selection of most perspective solutions
Objective function calculation
Stop criterion is obtained
No
Yes
Selection of the solution with the best objective function value
End

Fig. 2. The genetic algorithm of similarity estimation

The first step of the GA is to generate initial parameters of estimation model elements (population size and chromosome length) and to input values of weight coefficient and probability of crossover and mutation operators. Then, the initial population is to be formed on the basis of available learning data from the set C = {c_i} which semantic meta descriptions m(c_i) = {t_1, t_2, …, t_n(i)} was created for. Each chromosome element (gene) is a triplet in logical representation of the concept c_i. To estimate the fitness of each chromosome authors propose to calculate objective function value (4).

The chromosomes selection is carried out by determined method with the use of elitist strategy and partial substitution the least fitted chromosomes by the best fitted ones in terms of saving population size [19]. To generate new specimen set for each pair of selected parent chromosomes it is required to use crossover and mutation operators with pre-assigned probability. The crossover is carried out in a random way with the probability *Pc*. Crossing point is determined in random way within the assigned interval.

The mutation procedure is carried out with the child population obtained as the result of crossover and involves change of gene value by means of randomly selected number from the interval [0, 1] with the probability *Pm*.

The selection of the most perspective solutions is carried out on the basis of probabilities *Pv*, calculated for each population individual with the use of proportional selection [11–13]. After calculation the each chromosome fitness using the formula (4) and the selection of the best one it is required to decide whether to continue the evolutionary procedure of next generation creation or to end the learning procedure. The higher the objective function value is, the higher is the chromosome fitness. The GA work stops under one of following conditions:

(1) if the function *F* obtained expected value;
(2) if the assigned number of iterations (generations) does not improve already obtained valued of the *F*;
(3) if the time allotted for the problem solution is up.

Premature stop of the GA work can occur in case of population degeneration, which means the reduction of chromosomes diversity. The extreme form of degeneration is the condition, when all individuals have identical chromosomes [11].

As a result of the process of artificial evolution including the selection, the crossover, the mutation, the chromosome selection the quality of solutions in population gradually improves.

5 Experimental Research

Computational experiments were carried out for the purpose of the developed GA effectiveness research. To estimate the developed algorithm authors made a comparative analysis with the algorithm based on similarity measure TF-IDF. In the context of the measure each triplet is considered as an individual concept. To estimate the similarity the method uses cosine measure [5, 18] and the MKNN (Mutual KNN) algorithm providing the method of similarity estimation between nearest neighbor (concept triplets and relations) [16].

To carry out experimental research of the developed algorithm effectiveness authors designed software allowing us to implement the iterative procedure of semantic similarity estimation method parameters setting. Experimental research results allowed us to determine the dependence of algorithm execution time on input parameters of the similarity model: n is a number of chromosomes in population; the chromosome is represented as $m(c_i) = \{t_1, t_2, \ldots, t_n(i)\}$, where $n(i)$ is a number of triplets in logical representation of the concept c_i.

The dependences of developed algorithm execution time and TF-IDF and MKNN algorithms on number of similarity estimation model input data are shown on the Fig. 3. Time complexity of the developed algorithm is $O(n^2)$.

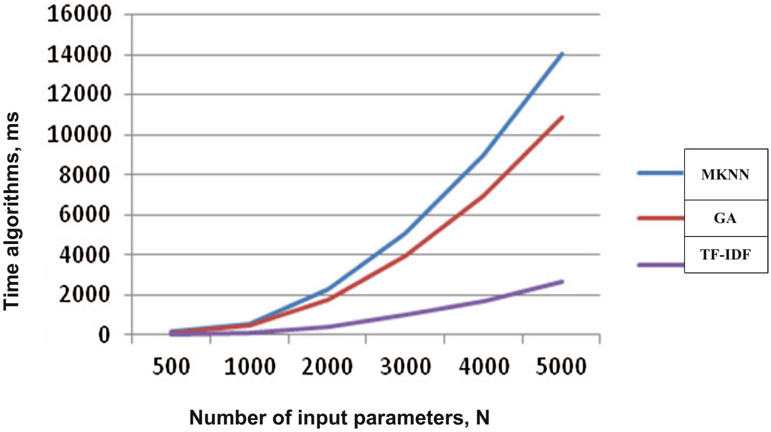


Fig. 3. The dependences of algorithms time on number of input parameters

Authors carried out a set of experiments in terms of completeness and accuracy of relevant concept extraction performed by the MKNN algorithms and the genetic algorithm with the use of two previously described similarity metrics and different number of ontology elements – triplets C_k represented by their meta descriptions. Results show almost linear growth of number of obtained concepts in dependence on the parameter k for both of algorithms (Fig. 4).

GA extracts more relevant concepts in terms of completeness and accuracy of the query. The reason is that the MKNN algorithm excludes pairs of elements that are not nearest neighbors [11].

During the GA search process rules of comparison of ontology elements triplets and query triplets are to be used. Performance accuracy depends on the quality of effective solutions obtained by the GA after each iteration, weighting results of criteria definition of knowledge elements similarity in ontology.

Proposed combined method represents the original mechanism of semantic search, which uses the GA to estimate similarity between ontology elements on the basis of the user's query description semantic metadata.

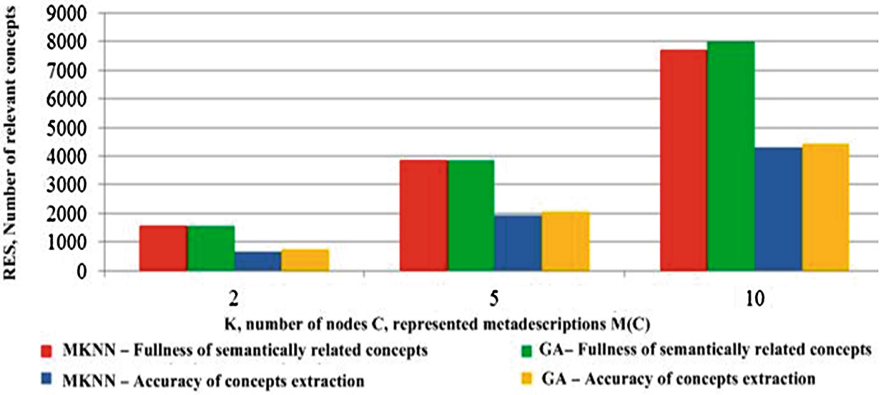


Fig. 4. Dependences of completeness and accuracy of concepts extraction

6 Conclusion

To determine measures of semantic closeness and coherence of problem-oriented knowledge elements authors developed the combined model of semantic similarity estimation which uses a set of interpreted taxonomical and associative dependences of meta descriptions represented in ontologies. The algorithm of semantic similarity estimation is based on evolutionary procedures and genetic optimum search operators which allows us to exclude non-informative and insignificant knowledge elements descriptions and to manage the speed of learning with the use of similarity threshold value assignment.

Acknowledgment. The study was performed by the grant from the Russian Science Foundation (project # 14-11-00242) in the Southern Federal University.

References

1. Bova, V.V., Kureichik, V.V., Zaruba, D.V.: Heuristic approach to model of corporate knowledge construction in information and analytical systems. In: 8th IEEE International Conference on Application of Information and Communication Technologies (AICT 2016), pp. 221–229. IEEE Press, Baku (2016)
2. Kravchenko, Y.A., Kuliev, E.V., Kursitys, I.O.: Information's semantic search, classification, structuring and integration objectives in the knowledge management context problems. In: 8th IEEE International Conference on Application of Information and Communication Technologies (AICT 2016), pp. 136–141. IEEE Press, Baku, Azerbaijan (2016)
3. Bova, V.V., Kureichik, V.V., Legebokov, A.A.: The integrated model of representation model of representation oriented knowledge in information systems. In: 8th IEEE International Conference on Application of Information and Communication Technologies (AICT 2014), pp. 111–115. IEEE Press, Astana (2014)

4. Kuliev, E.V., Kravchenko, Y.A., Kulieva, N.V., Kureichik, V.V.: Problem-oriented knowledge processing on the basis of hybrid approach. In: Proceedings of IEEE East-West Design & Test Symposium (EWDTS 2016), pp. 510–513, Yerevan, Armenia (2016)
5. Nguen, B.F., Tuzovskii, A.F.: Overview of semantic search approaches. In: Proceedings of Tomsk State University of Control Systems and Radio Electronics, vol. 2, pp. 234–237 (2010)
6. Penin, T., Wang, H., Tran, T., Yu, Y.: Snippet generation for semantic web search engines. In: Domingue, J., Anutariya, C. (eds.) ASWC 2008. LNCS, vol. 5367, pp. 493–507. Springer, Heidelberg (2008). doi:10.1007/978-3-540-89704-0_34
7. Knappe, R.: Measures of semantic similarity and relatedness for use in ontology-based information retrieval. Ph.D. thesis. Roskilde University, p. 143 (2006)
8. Bova, V.V.: Conceptual model of knowledge representation in the construction of intelligent information systems. In: Proceedings of SFU, vol. 156, pp. 109–117. TTI SFU, Taganrog (2014)
9. Kryukov, K.V., Pankova, L.A., Pronina, V.A., Shipilina, L.B.: Measures of semantic similarity in ontologies. J. Manage. Problems **2**, 2–14 (2010)
10. Tuzovskiy, A.F.: Working with ontologies in the knowledge management system the organization. In: Abstracts of the Second International Conference on Cognitive Science (CogSci-2006), pp. 581–583. SPb: SPbGU (2006)
11. Bova, V., Zaporozhets, D., Kureichik, V.: Integration and processing of problem-oriented knowledge based on evolutionary procedures. In: Abraham, A., Kovalev, S., Tarassov, V., Snášel, V. (eds.) Proceedings of the First International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'16). AISC, vol. 450, pp. 239–249. Springer, Cham (2016). doi:10.1007/978-3-319-33609-1_21
12. Rodzin, S., Rodzina, L.: Theory of bioinspired search for optimal solutions and its application for the processing of problem-oriented knowledge. In: 8th IEEE International Conference on Application of Information and Communication Technologies (AICT 2014), pp. 142–147. IEEE Press, Astana (2014)
13. Bova, V.V., Legebokov, A.A., Gladkov, L.A.: Problem-oriented algorithms of solutions search based on the methods of swarm intelligence. J. World Appl. Sci. J. **27**(9), 1201–1205 (2013)
14. Castano, S., Ferrara, A., Montanelli, S., Racca, G.: Semantic information interoperability in open networked systems. In: Bouzeghoub, M., Goble, C., Kashyap, V., Spaccapietra, S. (eds.) ICSNW 2004. LNCS, vol. 3226, pp. 215–230. Springer, Heidelberg (2004). doi:10.1007/978-3-540-30145-5_13
15. Panchenko, A.: Technology of the automated thesaurus construction for Information Retrieval. J. Intell. Syst. Technol. **9**, 124–140 (2009)
16. Zhu, H., Zhong, J., Li, J., Yu, Y.: An approach for semantic search by matching RDF graphs. In: Proceedings LAIRS Conference, pp. 450–454 (2002)
17. Gladkov, L.A., Kravchenko, Y.A., Kureichik, V.V.: Evolutionary algorithm for extremal subsets comprehension in graphs. J. World Appl. Sci. J. **27**, 1212–1217 (2013)
18. Li, Y., Bandar, Z.A., McLean, D.: An approach for measuring semantic similarity between words using multiple information sources. IEEE Trans. Knowl. Data Eng. **15**(4), 871–882 (2003)
19. Bova, V.V., Kureichik, V.V., Zaruba, D.V.: Data and knowledge classification in intelligence informational systems by the evolutionary method. In: 6th International Conference on Cloud System and Big Data Engineering (Confluence), pp. 6–11, Noida, India (2016)
20. Zaporozhets, D.Y., Zaruba, D.V., Kureichik, V.V.: Hybrid bionic algorithms for solving problems of parametric optimization. J. World Appl. Sci. J. **23**, 1032–1036 (2013)

The Development of Genetic Algorithm for Semantic Similarity Estimation in Terms of Knowledge Management Problems

Yury Kravchenko(✉), Ilona Kursitys, and Victoria Bova

Southern Federal University, Rostov-on-Don, Russia
{krav-jura,vvbova}@yandex.ru, i.kursitys@mail.ru

Abstract. This article is devoted to the development of a new approach for semantic similarity estimation. The main problem in knowledge search field is the complexity of identification and usage of key information, which is increasing constantly. To solve this problem we propose to modify previously developed knowledge filter running on the basis of the semantic concepts taxonomy tree as a systematization of complex areas and hierarchical knowledge. The knowledge filter meta-model is supplemented by a semantic similarity estimation block to obtain the most appropriate results in the context of semantics. We analyzed the assigned problem and observed different ways of semantic similarity estimation. To solve the problem we propose the graph model containing components of ontology triplets. The semantic similarity formula is presented in this paper. To increase the efficiency we developed the genetic algorithm for semantic similarity estimation. Experiments carried out on benchmarks show the efficiency of developed approach.

Keywords: Knowledge management · Ontologies · Meta-model · Semantic similarity · Graph model · Genetic algorithm

1 Introduction

In modern world the process of the society development is characterized by a constantly growing role of information technologies in science, production, management and other life spheres. Information technologies are spread all round, which resulted to increase of generated, transferred and processed information scope. Therefore, the development of new ways of information storing, representation, formalization, systematization and automatic processing are relevant today together with universal knowledge bases development that can be used in different practical problems. Since systems which are able to extract information from the text without human intervention are of a great interest, new technologies are being developed to solve mentioned problems in accordance with occurring requirements.

Today the problem of knowledge search is investigated by specialists in the field of personal and corporate knowledge management. It allows us to reduce time and work spent for problems solving and decision making at work and in everyday life [1, 2]. The problem comes especially acute in terms of knowledge search in the Internet as it is

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_9

more difficult to extract needed information in constantly growing information scope [3].

The main problem in the field of knowledge search is the complexity of identification and usage of key information. One of the ways to solve this problem is the improvement of semantic modeling for interpretation and applying users' search profiles pursuing similar aims as prior data [4, 15, 16].

2 Semantic Similarity

Let us consider text semantics as its meaning, which the author intended to convey by symbols. Nevertheless, for computer systems the text meaning strictly depends on the context which it is defined and processed in. In terms of Semantic Web the best mean to represent semantics is ontology [5].

The benefits of using ontological models are:

- To provide the easiness of system development;
- To give the possibility to obtain logical conclusions rather than raw data on the basis of stored data as the query results;
- To provide the simplicity of complex relations modeling in comparison with databases;
- To give the possibility to use the shared terminology with clearly defined semantics, which allows us to integrate and use information from heterogeneous sources.
- To give the possibility to alter data dynamically [6, 19].

Let us consider the ontological model (ontology) as the sign system <*C, P, I, L, T*>, where:

- *C* denotes a set of concepts;
- *P* denotes a set of properties (double predicate);
- *I* denotes a set of concept examples;
- *L* denotes a set of text labels or values of concepts and properties;
- *T* denotes partial order on *C* and *P*.

Similarity estimation between a document and a query is considered as a numeric value, which represents the degree of similarity between them. Similarity estimation is called semantic similarity estimation if and only if it is defined on the basis of document's and query's semantics.

In [7] authors researched different methods of semantic similarity estimation between terms in ontologies. There are several types of semantic similarity estimation as follows:

- taxonomical, which are based on hierarchical (generic, taxonomic) relations;
- relational, which are based on non-hierarchical (associative, problem-specific, 'horizontal') relations between ontology terms;
- attributive.

Methods based on hierarchical relations can be subdivided as follows:

- based on estimation of the shortest path (number of edges or vertexes) between vertexes in graph;
- based on estimation of taxonomy tree depth;
- based on the least common subsumer;
- based on common specificity of two vertexes.

Furthermore, there are restrictions imposed in those models, for instance, path configuration restrictions: path length and number of inflexions.

To estimate semantic similarity on the basis of non-hierarchical (associative) relations the comparison of two concepts with the third one and recursive clarification are used [7].

Attributive measure is estimated with the use of two concepts' common attribute values [8].

Hybrid methods are used to take into account all characteristics of two terms. In such case semantic similarity estimation consists of three parts–taxonomical, relational and attributive [8]:

$$S(i_1, i_2) = t \cdot S^t(i_1, i_2) + p \cdot S^p(i_1, i_2) + a \cdot S^a(i_1, i_2), \tag{1}$$

where i_1, i_2 denotes objects of semantic similarity estimation, *t, p, a* denotes coefficients determining 'weight' of each type of similarity in common measure, $t + p + a = 1$.

The main problem of most measures which are based on ontological structures is symmetry. Expert analysis shows that similarity measure is not always symmetrical for both hierarchical and attributive relations. The other relevant problem is semantic similarity estimation between ontological terms that are not related hierarchically, but have concrete problem-specific (horizontal or associative) relation.

The problem of semantic similarity estimation is particularly relevant in highly specialized subject areas. In the process of separation the specialized knowledge field from more general, many terms can be located too far from their generic terms and be a member of more general subject area in which the considered subject area is included. Thus, in specialized subject areas there are a lot of terms not having generic relations or their taxonomy has only one level (parent-child).

To solve problems mentioned above in [9] the authors proposed a graph model of semantic similarity estimation.

With the use of set of predicates P we can describe various relations between concepts and examples in ontologies. These relations are assigned on the basis of simple statements (triplets) <*s, p, o*>, where s and o denote subject and object of statement respectively, and p ∊ P denotes the predicate of ontology O [9].

Let us assume that each property of $p \in P$ can be assigned with the weight coefficient (semantic weight) $pv \in [0, 1]$, that specifies conceptual similarity between the subject and the object of a statement. If $pv = 1$, a subject and an object are semantically similar, if $pv = 0$, they are not. Coefficients assignment for predicates is made by developers in accordance with their understanding of ontology and requirements of problems being solved (Fig. 1).

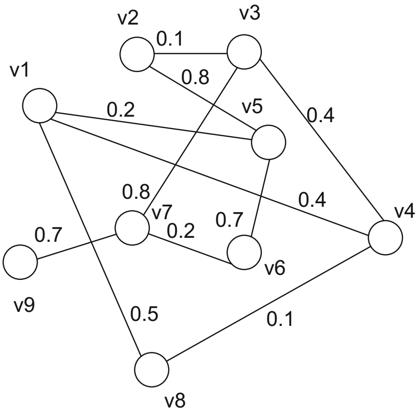


Fig. 1. Abstract graph model of triplet components

Let us construct a non-oriented graph *G* containing all subjects and objects of triplets in knowledge base in accordance with the following rules:

- to use only those triplets, which weight coefficients *pv* are nonzero ($pv \neq 0$);
- to consider subjects and objects of triplets as graph vertexes, and to assign to graph edges weights equal to coefficients *pv* values of predicate of a triplet which they are formed from;
- to assume, that inverted relation (on the basis of predicate <*owl:inverseOf*> between predicates *p1 (pv1)* and *p2 (pv2)* that adds in graph two edges weighted with *pv1* and *pv2*;
- to assume, that symmetrical relation adds in graph two edges with equal weights, for instance, <*owl:sameAs*> adds two edges with *pv* = *1.0* values (Fig. 1).

Let us put *Sim(α, β)* as the semantic similarity between elements α и *β*, where где α, $\beta \in C \cup I \cup P \cup T$. Considering the graph model mentioned above, semantic similarity is calculated as follows:

$$Sim(\alpha, \beta) = \max_{i=1 \to k}(Sim_{PATH_i}(\alpha, \beta)), \quad (2)$$

where *k* denotes a number of possible paths in graph *GO* from vertex α to vertex *β*.

Let us consider the path *PATH(α, β)* between vertexes α and *β* in graph *GO* as a set of edges (predicates) that leads from α to *β* in terms of their direction.

Semantic similarity between elements α and *β* in the direction of path *i* is calculated by the following formula:

$$Sim_{PATH_i}(\alpha, \beta) = \prod\nolimits_{j=1}^{h_i} pv_{i,j}, \quad (3)$$

where h_i denotes a number of semantic relations between α and *β* in the path *i*, $pv_{i,j}$ denotes the weight of edge on semantic predicate *j* in the path *i*.

Thus, the formula of semantic similarity estimation between ontology elements α and *β* is written as follows:

$$Sim(\alpha,\beta) = \max_{i=1\to k}(Sim_{PATH_i}(\alpha,\beta)) = \max_{i=1\to k}\left(\prod_{j=1}^{h_i} pv_{i,j}\right). \tag{4}$$

The value of *Sim(α, β)* are to have following properties:

- *Sim(α, β)* ∊ *[0,1];*
- *Sim(α, β) = 0* if there is no path from α to *β*;
- *Sim(α, α) = Sim(β, β) = 1.*

In exceptional cases *Sim(α, β)* value can possess value of 1, if there the relation between α and *β* is inverted.

To calculate the semantic similarity between vertex α and vertex β of graph G let us use the function *Sim(α, β)*, which returns the maximum value of similarity on possible paths between vertex α and vertex β. The function *Sim(α, β)* calls the function *PATH(α, β),* which in its turn calls the function *PATH(α, β, Path).*

The function *PATH(α, β, Path)* is recursive and it is called for each unpassed edge, that starts from vertex α. Therewith, the parameter α and the edge list on the path *Path* are changed.

The function *PATH(α, β, Path)* stops under following conditions:

(1) There is no maximum value of similarity on the new path *Path* (< *maxWeight*), as that the condition (*maxWeight* > 0) means that the path between initial vertex α and finite vertex β exists.
(2) Vertex α is passed (α ϵ PassedVertex).
(3) Vertexes α and β coincides.

Since the number of ontology components can become incalculably large, let us assume that the problem of semantic similarity estimation is NP-complete. There are no exact effective algorithms for its solution, and exhaustive enumeration can't solve it in polynomial time.

Genetic Algorithms (GA) has proven its performance in solving NP-complete problems. The efficiency of GA is especially shown in problems, which mathematical models are so complex, that appliance of standard optimization methods, such as branch-and-bound method, dynamic or linear programming methods is enormously difficult [10, 14].

3 Genetic Algorithm of Semantic Similarity Estimation

To improve the approach described in [9] we propose genetic algorithm (GA) of semantic similarity estimation.

In comparison with other optimization and search procedures genetic algorithm has following benefits:

- to carry the optimization process with a set of decisions rather than with single decision. This allows us to find new decisions on the basis of old ones, which are better of others. Thus, the properties of optimal decisions are developing;

- to provide the encoded structure of problem decision rather than a set of parameters. This increases the speed of data processing, i.e. the speed of optimization search process;
- to assign the rules of surviving in population for chromosome fitness estimation in addition with objective function calculation. This develops the diversity of population, which increases probability of finding the optimal decision;
- to use probabilistic rules in processes of population creation, crossover and mutation. This includes randomness in genetic search, which raises the prospect of local optimum overcoming [11, 13, 18].

Aside from features mentioned above, GA have following characteristics:

- rather wide range of application;
- possibility of combining GA with other methods, including non-evolutionary algorithms;
- the efficiency of GA in search for solutions in high dimension expansion;
- no restrictions for objective function form;
- clarity of GA schemes [12].

The developed genetic algorithm works with the graph model of triplet objects and subjects described in paragraph 2. The problem of semantic similarity estimation is to find the maximum product of weight coefficients on the path from vertex α to vertex β.

The chromosome has the form of vertex sequence, which represents the path from vertex α to vertex β. Chromosomes are generated randomly. Chromosomes' fitness is determined by the objective function.

The objective function is calculated in accordance with assigned weight coefficients of predicates on the principle of revealing vertex sequence, which provides the maximum product of coefficients. Let us assume $pv_1, pv_2, \ldots pv_n$ as weight coefficients of predicates, which are assigned by ontology development specialists. Thus, the objective function amounts to maximizing

$$\prod\nolimits_{i=1}^{n} pv_i \rightarrow max. \qquad (5)$$

To solve the problem of semantic similarity estimation with the use of GA, the following rules are to use [11, 13, 14]:

- to implement the crossover operator for genotypes of the fittest chromosomes with assigned probability. Then, one of children $\mathrm{P_i(t)}$ is chosen and saved as a member of new population with a probability of 0.5.
- to apply the inverse operator and the mutation operator to $P_i(t)$ with assigned probability. The obtained genotype is saved as $P_k(t)$.

A set of genetic operators are used for genetic algorithms. These are two-point, modified three-point and modified 'greedy' crossover operators, one-point mutation operator, modified mutation operator based on dichotomy technique, modified mutation operator with equally probable allocation and inversion operator (Fig. 2).

The stop criterion is obtaining the assigned number of populations.

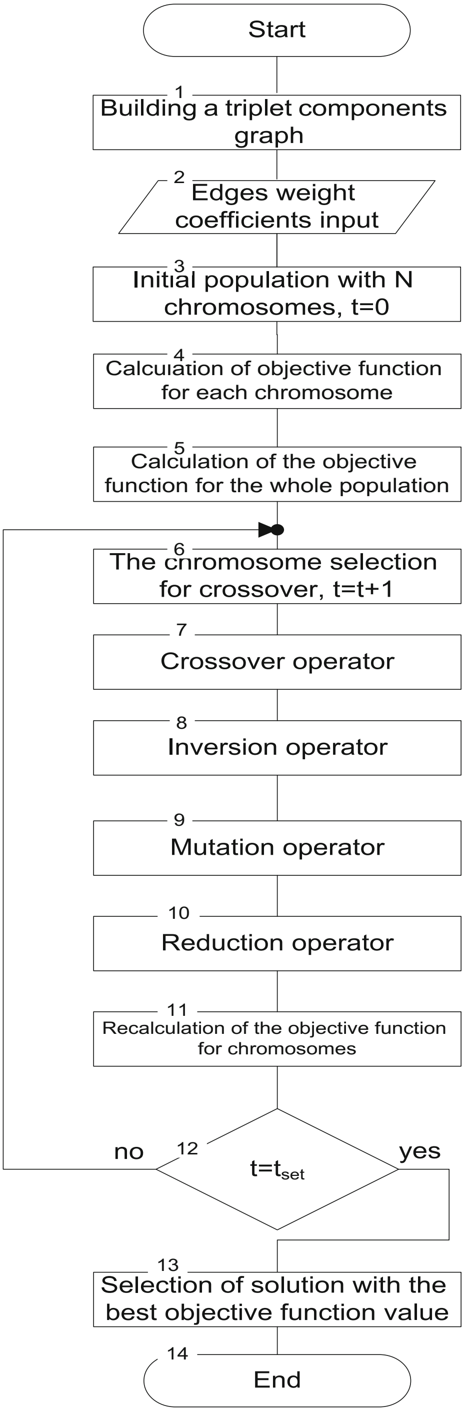

Start
1
Building a triplet components graph
2
Edges weight coefficients input
3
Initial population with N chromosomes, t=0
4
Calculation of objective function for each chromosome
5
Calculation of the objective function for the whole population
6
The chromosome selection for crossover, t=t+1
7
Crossover operator
8
Inversion operator
9
Mutation operator
10
Reduction operator
11
Recalculation of the objective function for chromosomes
12
t=t_set
no
yes
13
Selection of solution with the best objective function value
14
End

Fig. 2. Genetic Algorithm for semantic search estimation

4 Experimental Research

Experiments, carried out on different number of graph vertexes has shown that the time complexity of the algorithm is polynomial and represented as O(n2). The diagram of time complexity is shown on the Fig. 3.

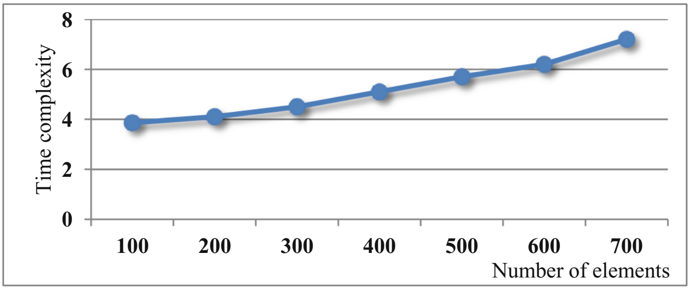


Fig. 3. The diagram of algorithm time complexity

The results of experiments carried out on different numbers of graph vertexes and edges are shown on Table 1. Experiments compared developed genetic algorithm and greedy algorithm of semantic similarity estimation.

As shown on Table 1, developed genetic algorithm is 29% more efficient than greedy algorithm on the average.

Table 1. The results of experiments

Graph of triplet components G(X,U)		Objective function value of Genetic Algorithm	Objective function value of Greedy Algorithm
Number of vertexes	Number of edges		
100	70	0,00003985	0,000028
200	90	0,000005893	0,0000042
300	110	0,00000065	0,00000045
400	130	0,0000000123	0,000000011
500	150	0,000000000485	0,0000000026
600	170	0,0000000000458	0,000000000000250
700	190	0,0000000000008523	0,00000000000000569

5 Conclusion

XXI century's society is inextricably connected with the development of information technologies. Over recent years the scope of generated, transferred and processed information has increased significantly. Therefore, the development of new ways of information storing, representation, formalization, systematization and automatic processing are relevant today. The common method for solving problems mentioned above is the usage of processed concepts semantics.

In studies concerning natural language semantics modeling the development of quantative methods of semantic information measurement in texts. One of the most important problems in this sphere is semantic similarity estimation (or estimation of inverted value–semantic distance) between lexicographic system units (dictionary, thesaurus, ontology).

In this paper we analyzed common ways for semantic similarity estimation on the basis of hierarchical and non-hierarchical relations. The graph model of ontology triplets' components is proposed, and the formula for semantic similarity calculation is reduced. We developed genetic algorithm working with the graph model, and proposed a set of genetic operators. Experiments show the efficiency of developed approach.

Acknowledgements. The study was performed by the grant from the Russian Science Foundation (project # 14-11-00242) in the Southern Federal University.

References

1. Dorsey, P.: Personal knowledge management [e-resource], 16 February 2016. http://www.360doc.com/content/05/1228/22/2563_51065.shtml
2. Martin, J.: Personal knowledge management: the basis of corporate and institutional knowledge management. In: Managing Knowledge: Case Studies in Innovation, vol. 6. University of Alberta, Faculty of Extension, Alberta (2000)
3. Madhu, G., Govardhan, A., Rajinikanth, T.V.: Intelligent semantic web search engines: a brief survey. Int. J. Web Semant. Technol. (IJWesT) **2**(1), January 2011
4. Kravchenko, Y., Kursitys, I., Bova, V.: Models for supporting of problem-oriented knowledge search and processing. In: Abraham, A., Kovalev, S., Tarassov, V., Snášel, V. (eds.) Proceedings of the First International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'16). AISC, vol. 450, pp. 287–295. Springer, Cham (2016). doi:10.1007/978-3-319-33609-1_26
5. Kerschberg, L., Jeong, H., Kim, W.: Emergent semantics in knowledge sifter: an evolutionary search agent based on semantic web services. In: Spaccapietra, S., Aberer, K., Cudré-Mauroux, P. (eds.) Journal on Data Semantics VI. LNCS, vol. 4090, pp. 187–209. Springer, Heidelberg (2006). doi:10.1007/11803034_9
6. Cui, Z., Damiani, E., Leida, M.: Benefits of ontologies in real time data access. In: Digital EcoSystems and Technologies Conference, DEST 2007, 21-23 February, pp. 392–397. Inaugural IEEE-IES (2007)
7. Slimani, T.: Article: description and evaluation of semantic similarity measures approaches. Int. J. Comput. Appl. **80**(10), 25–33 (2013)
8. Maedche, A., Zacharias, V.: Clustering ontology-based metadata in the semantic web. In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) PKDD 2002. LNCS, vol. 2431, pp. 348–360. Springer, Heidelberg (2002). doi:10.1007/3-540-45681-3_29
9. Le, H., Tuzovsky, A.F.: Development of semantic digital libraries on the basis of the ontological models. In: Selected Papers of the 15th All-Russian Scientific Conference Digital Libraries: Advanced Methods and Technologies, Digital Collections, RDCL 2013, Yaroslavl (2013)

10. Sochnev, A.: Resource allocation in production system using petri nets and genetic algorithm. UBS **39**, 238–253 (2012)
11. Gladkov, L.A., Kureichik, V.V., Kureichik, V.M.: Geneticheskie algoritmy (Genetic Algorithms). Fizmatlit, Moscow (2006)
12. de Araujo, S.A., Poldi, K.C., Smith, J.: A genetic algorithm for the one-dimensional cutting stock problem with setups. Pesqui. Oper. **34**(2), 165–187 (2014). http://www.scielo.br/scielo.php?Script=sci_arttext&pid=S0101-74382014000200165&lng=en&nrm=iso
13. Kureichik, V.V., Kureichik, V.M., Rodzin, S.I.: Theory of Evolutionary Computation, 260 p. Publishing Firm, Physical and Mathematical Literature, Moscow (2012)
14. Kureichik, V.M., Rodzin, S.I.: Evolutionary algorithms: genetic programming. J. Comput. Syst. Sci. Int. **41**(1), C. 123–C. 132 (2002)
15. Bova, V.V., Kravchenko, Y.A., Kureichik, V.V.: Development of distributed information systems: ontological approach. In: Silhavy, R., Senkerik, R., Oplatkova, Z.K., Prokopova, Z., Silhavy, P. (eds.) Software Engineering in Intelligent Systems. AISC, vol. 349, pp. 113–122. Springer, Cham (2015). doi:10.1007/978-3-319-18473-9_12
16. Kureichik, V.V., Kravchenko, Y.A., Bova, V.V.: Decision support systems for knowledge management. In: Silhavy, R., Senkerik, R., Oplatkova, Z.K., Prokopova, Z., Silhavy, P. (eds.) Software Engineering in Intelligent Systems. AISC, vol. 349, pp. 123–130. Springer, Cham (2015). doi:10.1007/978-3-319-18473-9_13
17. Kravchenko, Y.A., Kureichik, V.V.: Knowledge management based on multi-agent simulation in informational systems. In: 8th IEEE International Conference Application of Information and Communication Technologies, AICT 2014, pp. 264–267 (2014)
18. Zaporozhets, D.U., Zaruba, D.V., Kureichik, V.V.: Hybrid bionic algorithms for solving problems of parametric optimization. World Appl. Sci. J. **23**(8), 1032–1036 (2013)
19. Fishwick, P.A., Miller, J.A.: Ontologies for modeling and simulation: issues and approaches. In: Proceedings of Winter Simulation Conference, pp. 259–264 (2004)
20. Tuzovsky, A.F., Yampolsky V.Z.: Knowledge management system, as new maturity level of company information system. In: Proceedings of KORUS 2002, Novosibirsk, pp. 145–147 (2002)

Scheme Partitioning by Means of Evolutional Procedures Using Symbolic Solution Representation

Oleg B. Lebedev(✉), Svetlana V. Kirilchik, and Eugeniy Y. Kosenko

Southern Federal University, Rostov-on-Don, Russia
{lebedev.ob,kirilchik}@mail.ru, eykosenko@gmail.com

Abstract. The paper provides a methodology of symbolic representation for solving the partitioning problem. The approach is based on adjacency matrix of a graph, adaptive mechanisms for adjacency matrix modification. Also the structure of adjacency matrix evolutionary modification for solving the problem of finding a partition is considered.

Keywords: Partitioning · Graph · Hyper-graph · Evolutionary procedures · Adaptation · Topology · Matrix · Solution symbolic representation

1 Introduction

One of the most relevant integer programming tasks is partitioning, considered in combinatorial part of the graph theory. Modern VLSI contains tens of millions of transistors, so due to the limited possibilities of the computing resources (memory, speed) the entire scheme topology can not be designed. Normal scheme partitioning is implemented by grouping components into the blocks. The partitioning results in variety of blocks and interconnections between the blocks [1].

A hierarchical partitioning structure is applied to large schemes. By now most of the developed partitioning algorithms use graph or hypergraph as a scheme model. Graph partitioning procedure is part of a large number of algorithms solving various problems. Often, this procedure is implemented in iterative structures [2]. This requires high quality and time standards to the problem of finding the maximal matching.

Existing to date a greater number of partitioning algorithms provide acceptable results obtained when solving the problems of low and medium complexity [3]. The needs for the problems of large and very large dimension appear to be the motivation to research and develop new effective algorithms. Analysis of the literature shows that the most successful in these circumstances are the methods based on simulation of evolutionary processes [4].

2 Main Provisions

The partitioning of the hypergraph with the weighted vertices and edges is formulated the following way [5].

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_10

Let $H = (X, E)$ be given a *hypergraph*, where $X = \{x_i \mid i = 1, 2, ..., n\}$ is the set of vertices and $E = \{e_j \mid e_j \subset X, j = 1, 2, ..., m\}$ is a set of edges (each edge links a subset of vertices). Let $\Phi = \{\varphi_i \mid i = 1, 2, ..., n\}$ be given a set of vertice weights and $\Psi = \{\psi_i \mid i = 1, 2, ..., n\}$ be given a set of edge weights. It is necessary to form K - nodes, that is the set X divided into K non-empty and disjoint subsets $X_v, X_v, X = \cup X_v$, $(\forall\ i,j)\ [X_i \cap X_j = \varnothing], X_v \neq \varnothing$.

Restrictions are applied to the formed nodes. Vector $P = \{p_v \mid v = 1, 2, ..., k\}$ defines maximal total weight of vertices assigned to node v, and vector $N = \{n_v \mid v = 1, 2, ..., k\}$ designates maximal number of vertices assigned to node v.

The capacity restrictions are:

$$\sum\nolimits_{i \in I} \varphi_i \leq p_v, I = \{i | x_i \in X_v\}, v = 1, 2, \ldots, k \tag{1}$$

$$|X_v| \leq n_v, v = 1, 2, \ldots, k \tag{2}$$

Equation (1) is a restriction on the maximal weight of the node, and the Eq. (2) - on the maximal number of vertices in the node [6].

Sometimes the number of outputs γ_{max} for the nodes is given. The restriction takes the form of:

$$\begin{array}{c} \gamma_v \leq \gamma_{max}, v = 1, 2, \ldots, k \\ \gamma_v = |E_v|, E_v = \{e_j | (e_j \cap X_v \neq \emptyset) \& (e_j \cap X_v \neq e_j)\} \end{array} \tag{3}$$

E_v is the set of edges connecting set of vertices X_v with the vertices of the rest nodes.

The main criterion is F_1 that denotes the total cost of edges in the cut.

$$F_1 = \sum\nolimits_{j=J} \varphi_j, J = \{j | e_j \in \mathrm{C}\} \tag{4}$$

$C = \{e_j \mid (\forall v)\ [e_j \cap X_v \neq e_j]\}$ – the set of edges in the cut.

Second frequently used criterion is F_2 - the total number of outputs [7].

$$F_2 = \sum\nolimits_{v=1}^{v} \gamma_v$$

The criterion F can be also used, it is an additive convolution of criteria F_1 and F_2.

$$F = k_1 \cdot F_1 + k_2 \cdot F_2$$

Here is an example. Let $G = (X, U)$ be given a graph, shown in Fig. 1.

The task is to form three nodes. Considered legal number of vertices assigned to nodes is $n_1 = 3, n_2 = 4, n_3 = 3$. Initial partitioning $X_1 = \{x_1, x_2, x_3\}$, $X_2 = \{x_4, x_{10}, x_7, x_6\}$, $X_3 = \{x_8, x_5, x_9,\}$ is given [8]. The number of connections between nodes set to 4. The state of object to be optimized is estimated by vector $S = \{s_i \mid i = 1, 2, ..., n\}$. The value s_i appears to be the serial number of a vertice x_i. Elements s_i that satisfy $1 \leq i \leq n_i$ correspond to the first node X_1. Elements s_i that satisfy $n_1 + 1 \leq i \leq n_1 + n_2$

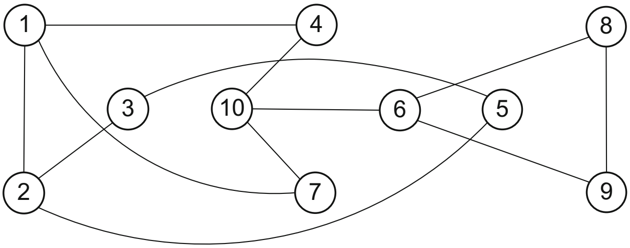


Fig. 1. Initial partitioning of graph G

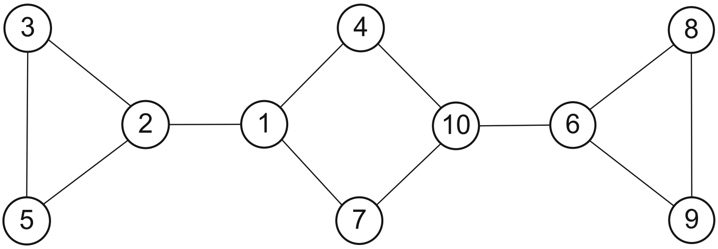


Fig. 2. Optimal partitioning of graph G

correspond to the second node X_2 and so on. In our case, $S_н = \{x_1, x_{2,} x_{3,}, x_4, x_{10,} x_7, x_6, x_8, x_{5,} x_{9}\}$ [9].

Optimal partitioning of graph G, described by the vector $S_o = \{x_2, x_{3,} x_{5,}, x_1, x_{4,} x_7, x_{10}, x_6, x_{8,} x_9\ \}$ is shown in Fig. 2

Each decision (each vector S) corresponds to adjacency matrix R which rows and columns are labeled with nodes of graph G in the same order as in corresponding vertices are located in S. Adjacency matrixes for vectors $S_н$ and S_o are shown in Figs. 3 and 4. Since the adjacency matrix is symmetrical about the main diagonal, in the figures the top part of the matrix is not filled in and is not considered [10]. Thus, the rows and columns from 1 to n_1 are designated by serial numbers of vertices belonging to the first node, and the rows and columns from $((n_1 + n_2 +\ldots + n_i) + 1)$ to $((n_1 + n_2 + \ldots + n_i) + n_{i+1}$ + ni + 1) - belonging to (n_{i+1}) node [11].

Let the columns and rows of the matrix with the numbers from l to $l + m$ are designated by elements belonging to the one X_i node. Now we'll consider area Q_i of matrix R formed by the intersection of columns and rows with the numbers from l to $(l + m)$ symmetrically about the main diagonal [12]. Elements of the range Q_i of matrix R reflect the connections between the corresponding vertices of the node X_i. The number h_i of nonzero elements of the area Q_i of matrix R equals the number of internal connections between the vertices of node X_i.

The number F of non-zero elements of matrix R that do not belong to any range Q_i equals the number of external connections between the nodes X_i. Figures 3 and 4 show areas Q_1, Q_2, Q_3 formed respectively at the intersection of 1–3, 4–7 and 8–10 rows and columns of the adjacency matrix. Let *NQ* be the elements set of matrix *R* not belonging to any area Q_i.

Suppose that we call *NQ* external relations range. For a matrix shown in Fig. 3 $Q_1 = 2, Q_2 = 3, Q_3 = 1, F = 6,$ for matrix in Fig. 4 $Q_1 = 3, Q_2 = 4, Q_3 = 3, F = 2.$ The graph partitioning into nodes is made for the purpose of minimization of the number of connections between the nodes [13]. The problem is form an area of external

	1	2	3	4	10	7	6	8	9	5
1	⊗									
2	1	⊗								
3		1	⊗							
4	1			⊗						
10				1	⊗					
7	1				1	⊗				
6					1		⊗			
8							1	⊗		
9							1	1	⊗	
5		1	1							⊗

Fig. 3. Initial state of adjacency matrix

	2	3	5	1	4	7	10	6	8	9
2	⊗									
3	1	⊗								
5	1	1	⊗							
1	1			⊗						
4				1	⊗					
7				1		⊗				
10					1	1	⊗			
6							1	⊗		
8								1	⊗	
9								1	1	⊗

Fig. 4. Final state of adjacency matrix

connections *NQ* with a minimal value *F* by means of rearranging rows and columns in the adjacency matrix of the graph [15].

3 Evolutionary Mechanisms to Form a Minimal Area of External Relations

Area of external relations *NQ* with the minimal value F in matrix R is formed within its evolutionary modifications. The evolutionary modification of the matrix *R* is carried out by means of selective group permutations of adjacent rows and columns. It provides a directional consistent movement of elements of the matrix *R* with nonzero values from the area of external relations *NQ* to areas *Qi*. Adaptive process consists of repeated steps, each of them is a transition from one solution (state of matrix *R*) to the best one [17].

At each step pairs of adjacent rows and columns *(i, i + 1)* are analyzed. The analysis is carried out in two strokes. In the first stroke all pairs *(i, i + 1)* where the first element *i* is an odd number are analyzed. In the second stroke all pairs where the first element *i* is even are analyzed.

For example, let n = 10, then in the first stroke pairs of rows and columns {(1,2), (3,4), (5,6), (7,8), (9,10)} are considered. In the second stroke pairs {(2,3), (4,5), (6,7), (8,9)} are considered.

Pairs of rows and columns are analyzed independently. According to the analysis the decision to interchange adjacent pair of rows and columns is made.

Local permutations goal is to move the non-zero elements of the matrix from bottom to up and from the right to the left. The global objective is to form an area of external connections *NQ* with the minimal value *F*, that is a partitioning of graph *G* with the minimal number of connections between the nodes [17].

The pair of rows and columns *(l, l + 1)* in *n*n* matrix $R = ||r_{ij}||$ is selected for analysis. And let the rows and columns intersect the area Q_k, formed at the intersection of columns and rows with numbers from *v* to *w*. According to the following formulas parameters S_1, S_2, S_3, S_4 are calculated.

$$\sum_{j=1}^{v-1} r_{lj} = S_1; \quad \sum_{j=1}^{v-1} r_{l+1,j} = S_2.$$

$$\sum_{i=w+1}^{n} r_{il} = S_3; \quad \cdot \quad \sum_{i=w+1}^{n} r_{i,l+1} = S_4.$$

S_1 and S_2 are the sums of *l-th* and *l + 1-th* raws in matrix *R*, the elements do not belong to the area Q_k.

S_3 and S_4 are the sums of *l-th* and *l + 1-th* columns in matrix *R*, the elements do not belong to the area Q_k.

If a pair of raws *(l, l + 1)* in matrix *R* belong to two adjacent areas Q_k and Q_{k+1}, the parameters S_1, S_2, S_3, S_4 are calculated as follows:

$$\sum_{j=1}^{l-1} r_{lj} = S_1; \quad \sum_{j=1}^{l-1} r_{l+1,j} = S_2$$

$$\sum_{i=l+2}^{n} r_{il} = s_3; \quad \sum_{i=l+2}^{n} r_{i,l+1} = S_4$$

In this case, the sums S_1 and $S_{2,}$ S_3 and S_4 include all the elements of *l-th* and *(l + 1)-th* rows, *l-th* and *(l +1)th* columns of the triangular matrix *R*, except element $r_{l+1,l}$.

The main idea of the analysis is to determine the truth value of the 2 following conditions.

1. $(S_2 - S_1) + (S_3 - S_4) > 0.$
2. $(S_2 - S_1) + (S_3 - S_4) = 0.$

The answer is a qualified "yes", that is - to rearrange, answer is generated if the condition 1 holds. In the case the condition 2 is satisfied the answer "yes" is generated with the a priori probability *P*. Answer "No" is generated in all the other cases [13].

Adaptive search procedure continues until there are pairs, for which conditions 1 and 2 hold. As a result, the area of external connections *NQ* with the minimal value *F* will be formed and the graph partitioning with the minimal number of connections between nodes will be defined [18].

Example. Figure 1 shows graph *G*. Initial partitioning is given. The adjacency matrix of a graph *G* is shown in Fig. 3.

At the first step and first stroke pairs {(1,2), (3,4), (5,6), (7,8), (9,10)} are considered, at the second stroke - pairs {(2 3), (4,5), (6,7), (8,9)}. Pair of rows and columns is rearranged, if one of the above two conditions is fulfilled. In the initial matrix *R* columns and rows are labeled with serial numbers of vertices of the graph G. Swap of adjacent pair of rows and columns *(i, i + 1)* also leads to a rearrangement of its labels. Further rows and columns will be identified by its labels [19].

Step 1, stroke 1: (1.2) - yes; (3.4) - no; (10.7), yes; (6.8) - no; (9.5), yes. Thus, the rearrangement is carried out on pairs (1,2), (10,7), (9.5) at the 1st stroke of the 1st step. The modified matrix *R* shown in Fig. 5.

Step 1, stroke 2: (1.3) - yes; (8.5) - yes. The modified matrix is shown in Fig. 6.
Step 2, stroke 1: (6.5) - yes. The modified matrix *R* is shown in Fig. 7.
Step 2, stroke 2: (10.5) - yes. The modified matrix *R* is shown in Fig. 8.
Step 3, stroke 1: (7.5) - yes. The modified matrix *R* is shown in Fig. 9.
Step 3, stroke 2: (4.5) - yes. The modified matrix *R* is shown in Fig. 10.
Step 4, stroke 1: (1.5) - yes. The modified matrix *R* is shown in Fig. 4.
Step 4, stroke 2: no permitted permutations.

After the four steps in the modified matrix the area of external relations *NQ* with a minimal value *F = 2* is formed [20, 21].

To overcome the local barrier approaches based on a combination of different types of evolution are used.

The first approach uses the idea of annealing simulation method. In case the analysis shows that conditions 1,2,3 are not met, the rearrangement is performed with

	2	1	3	4	7	10	6	8	5	9
2	⊗									
1	1	⊗								
3	1		⊗							
4		1		⊗						
7		1			⊗					
10				1	1	⊗				
6						1	⊗			
8							1	⊗		
5	1		1						⊗	
9							1	1		⊗

Fig. 5. Step 1, stroke 1

	2	3	1	4	7	10	6	5	8	9
2	⊗									
3	1	⊗								
1	1		⊗							
4			1	⊗						
7			1		⊗					
10				1	1	⊗				
6						1	⊗			
5	1	1						⊗		
8							1		⊗	
9							1		1	⊗

Fig. 6. Step 1, stroke 2

the probability $P = exp(-\Delta F/kT)$, where T denotes the temperature, ΔF - the difference in the sums of the analyzed raws.

	2	3	1	4	7	10	5	6	8	9
2	⊗									
3	1	⊗								
1	1		⊗							
4			1	⊗						
7			1		⊗					
10				1	1	⊗				
5	1	1					⊗			
6						1		⊗		
8								1	⊗	
9								1	1	⊗

Fig. 7. Step 2, stroke 1

	2	3	1	4	7	5	10	6	8	9
2	⊗									
3	1	⊗								
1	1		⊗							
4			1	⊗						
7			1		⊗					
5	1	1				⊗				
10				1	1		⊗			
6							1	⊗		
8								1	⊗	
9								1	1	⊗

Fig. 8. Step 2, stroke 2

The second approach uses one of the genetic search structures. The population appears to be the set of adjacent matrixes (encoded as chromosomes). Decoding, that is obtaining solutions, is carried out by the described above adaptive process.

Time complexity of adaptive procedures in one step is *O(n)*. Comparison with the known algorithms has shown that the betters results are obtained in less time.

	2	3	1	4	5	7	10	6	8	9
2	⊗									
3	1	⊗								
1	1		⊗							
4			1	⊗						
5	1	1			⊗					
7			1			⊗				
10				1		1	⊗			
6							1	⊗		
8								1	⊗	
9								1	1	⊗

Fig. 9. Step 3, stroke 1

	2	3	1	5	4	7	10	6	8	9
2	⊗									
3	1	⊗								
1	1		⊗							
5	1	1		⊗						
4			1		⊗					
7			1			⊗				
10					1	1	⊗			
6							1	⊗		
8								1	⊗	
9								1	1	⊗

Fig. 10. Step 3, stroke 2

4 Conclusion

Due to the fact that the partitioning problem considered in the scope of combinatorial graph theory, is one of the most relevant integer programming tasks, an algorithm was developed to solve this problem.

It is known that a hierarchical partitioning structure is applied to large schemes.

To set partitioning problem hypergraph was used as a scheme model.

Graph partitioning procedure is part of a large number of algorithms to solve various problems. This procedure is quite often implemented in iterative structures as the high demands are made to the quality and the time to solve the problem of finding the maximal matching.

Paper shows as the state of the optimization object is estimated by vector S, which is associated with the adjacency matrix *R*.

The developed procedure has allowed to simplify the problem of graph partitioning into nodes with the minimal number of interconnections. The goal is to form the external connections area with the minimal criterion value F by means of rows and columns permutations in the adjacency matrix.

The evolutionary modification of the formation process of external connections with the minimal criterion value in the matrix *R*. The modification point is to permutate adjacent rows and columns of the matrix *R*, which provides a consistent directional movement of the elements within the matrix.

Implemented adaptive process consists of repeated steps, each of that is a transition from one solution to the better one and is performed in two strokes.

Experimental studies have been carried out to prove the high efficiency of the proposed evolutionary procedures.

Acknowledgements. This research is supported by grants of the Russian Foundation for Basic Research of the Russian Federation, the project № 15-01-05297.

References

1. Lebedev, B.K.: Adaptation in CAD. In: Monograph. TRTU Publishing House, Taganrog (1999)
2. Lebedev, B.K.: Methods of search adaptation for VLSI CAD. In: Monograph. TRTU Publishing House, Taganrog (2000)
3. Lebedev, B.K.: Ant partitioning algorithms using non-canonical task representation. Reporter of the Rostov State University of Railway Connections, no. 3(63), pp. 42–47. RSURC Publishing House (2016)
4. Lebedev, B.K., Lebedeva, E.M.: Partitioning into classes by means of alternative collective adaptation. Izvestiya of SFU. Eng. Sci. **7**(180), 89–101 (2016). SFU publishing house, Rostov-on-Don
5. Lebedev B.K., Lebedev V.B. Adaptive bee colony behavior model-based program for solving graph problems. Certificate of state registration of the computer program no. 2014663152 issued on 02 April 2015
6. Lebedev, B.K., Kovalenko, M.S.: Solution for partitioning problem based on search engine adaptation. In: Proceedings of the Congress on Intelligent Systems and Information Technology, vol. 3, pp. 72–83. SFU Publishing House, Taganrog (2015). Scientific edition in 3 volumes
7. Lebedev, B.K.: Nature-inspired VLSI design methods. In: Monograph. LAP LAMBERT Academic Publishing GmbH & Co. KG Heinrich – Bocking- Str. 6–8, 66121 Saarbrucken, Deutschland (2014)

8. Lebedeva, E.M.: Scheme partitioning based on ant colony method. Electron. J. Inform. Comput. Sci. Eng. Educ. **2**(13), 20–26 (2013). TTI SFU Publishing House, Taganrog
9. Models of adaptive ant colony behavior in the design tasks. In: Monograph. TTI SFU Publishing House, Taganrog (2013)
10. Gladkov, L.A., Kureichik, V.V., Kureichik, V.M., Lebedev, B.K., Lebedev, V.B., Nuzhnov, E.V., Rodzin, S.I.: Elements of the evolutionary optimization and decision-making theory based on nature-inspired methods. In: Monograph. SFU Publishing House, Taganrog (2013)
11. Zhilenkov, M.: EVA scheme partitioning by mean of ant colony method. In: Proceedings of the 59th Student Conference, pp. 17–18. TTI SFU Publishing House, Taganrog (2012)
12. Kureichik, V.M., Lebedev, B.K.: Hybrid partitioning algorithm based on natural decision-making mechanisms. In: Artificial Intelligence and Decision-Making, pp. 3–15. Publishing House of the Institute of System Analysis, RAS, Moscow (2012)
13. Kureichik, V.M., Lebedev, B.K.: Adaptation applied to topology design problems. In: Monograph. LAP LAMBERT Academic Publishing Gmbh & Co. KG, Saarbrucken, Germany (2012)
14. Gladkov, L.A., Kureichik, V.V., Kureichik, V.M., Lebedev, B.K., Lebedev, V.B.: New approaches and technologies to build decision-making algorithms for optimization problems. In: Collective Monograph. TTI SFU Publishing House, Taganrog (2011)
15. Lebedev, B.K.: Intelligent VLSI topology synthesis procedures. TTI SFU Publishing House, Taganrog (2003)
16. Dorigo, M., Stützle, T.: Ant colony optimization: overview and recent advances. In: Gendreau, M., Potvin, Y. (eds.) Handbook of Metaheuristics. International Series in Operations Research and Management Science, vol. 146, pp. 227–263. Springer, New York (2010). 2nd edn.
17. Dorigo, M., Maniezzo, V., Colorni, A.: The ant system: optimization by a colony of cooperating agents. IEEE Trans. Syst. Man Cybern.-Part B **26**(1), 29–41 (1996)
18. Cong, J., Wu, C.: Global clustering-based performance-driven circuit partitioning. In: Proceedings of ISPD (2002)
19. Engelbrecht, A.P.: Fundamentals of Computational Swarm Intelligence. Wiley, Chichester (2005)
20. Mazumder, P., Rudnick, E.: Genetic Algorithm For VLSI Design, Layout & Test Automation. Pearson Education, Bengaluru (2003)
21. Poli, R.: Analysis of the publications on the applications of particle swarm optimisation. J. Artif. Evol. Appl. 10 p. Article ID 685175 (2008)

Application of Ontological Approach for Learning Paths Formation

Boris K. Lebedev, Oleg B. Lebedev(✉), and Tatiana Y. Kudryakova

Southern Federal University, Rostov-on-Don, Russia
lebedev.b.k@gmail.com, tati.kudryakova@gmail.com,
lebedev.ob@mail.ru

Abstract. Provided algorithm allows to generate variants of learning paths that include subsets of small modules of educational material. Learning paths optimization method based on the ontological approach is considered. To form the optimal learning path algorithm is proposed. an algorithm Ant colony behavior is used as a model. The distinctive feature is a graphical representation of the learning path suitable for analysis of obtained results.

Keywords: Educational environment · Ontology · Model · Learning path · Optimization · Swarm intelligence · Ant colony

1 Introduction

The basis of learning process in open education is a purposeful, monitored, intensive individual work of trainee who can study at convenient place, according to individual schedule with a set of special teaching aids. Currently, within the frame of semantic technologies, ontological approach to presentation of subject area knowledge is actively studied and developed. Intelligent information systems are developed based on the approach, including knowledge portals designed for effective knowledge sharing and interaction during intellectual distributed team work [1]. Ability to adapt to specific teaching tasks, level of competency and personal characteristics of a trainee is the main requirement to modern learning resources used in the teaching knowledge management systems. The task is solved with the help of modularity of electronic learning resources which allows easy creating of learning paths afterwards [2]. To eliminate the lack of coherence between autonomous modules, integrating knowledge management environment is used. The environment is based on the subject ontologies which are a complex of notions in knowledge area and relations between them, including methods for interpretation of notions and relations. Ontology is a model of knowledge representation which can be used for describing semantics of objects of subject-oriented information systems. In this case relations between notions are expressed by arcs-references of vertexes-notions (concepts) of ontology. The required contents of learning modules can be based on navigation in semantic net [2, 3]. Navigation in notion semantic net is based on creation of simulation graph model which uses a key word expressing user request and list of notions connected with the current notion and forming its environment at the beginning. The analyzed notions assist in creating a list

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_11

of learning modules with the current notion [4, 5]. Within this approach the setting is performed due to selection of navigation paths in semantic net and decision-making concerning selection of concept at each vertex of network. The work [4] described formal method of creating individual learning paths based on the model of educational space in ontological bases of educational resources presented in the form of AND/OR graph model. The major drawback of the proposed approach is practically manual mode of full or partial search of paths for the best one in concept semantic net based on minimization of Boolean functions [3]. A swarm algorithm for optimal learning path is proposed in the work for making decisions close to optimal.

2 Model of Educational Space in Ontological Bases of Educational Resources

We will use terms and definitions from the work by Norenkov I.P., Sokolov N.K. [4] for describing the model of educational space and synthesis method for textbooks.

Target concept – concept (notion) which shall be studied with the help of synthesized textbook.
Initial concept – concept which has been studied by user.
Concept semantic net – network with concepts corresponding to net vertexes and relations between them corresponding to arcs. Thus, if X is a determinative notion, Y is a notion to be determined, then the arc goes from vertex V to vertex Y.
Module semantic net – network with modules corresponding to net vertexes. Arc X -> Y is possible, if a notion is determined in module X and used in module Y.

A notion determined in a module is called module output notion and notions used in the module for explanation of the output notion are called module input notions.

Educational space model is AND-OR graph $G = (M \cup K,U)$, where M – set of modules, K – set of concepts, created by integration of semantic nets of concepts and modules.

Module example is given in Fig. 1 from work [4], where concept vertexes are showed in the form of ovals, and module vertexes – in the form of rectangles. Note that disjunction connective corresponds to concept vertexes and that's why they are called OR vertexes, and conjunction connective corresponds to module vertexes and they vertexes are called AND vertexes.

Algorithm of Learning Path Synthesis. Learning path synthesis starts with selecting a set of target and initial concepts. The selection is made by tutor considering trainee personal characteristics and/or course education program. In case of self-education the trainee, who needs a textbook, sets targets alone.

The proposed algorithm is based on procedure "Individual path" selected in *educational space model* (AND-OR graph G) of subgraph $G^* \subset G$, including individual learning path.

Learning path is calculated by propagating a wave to *educational space model*. The wave is propagated by successive displacement of front of vertexes Φ_i on graph G. The front is created and moved step by step. Front Φ_0^k including a set of all target concepts

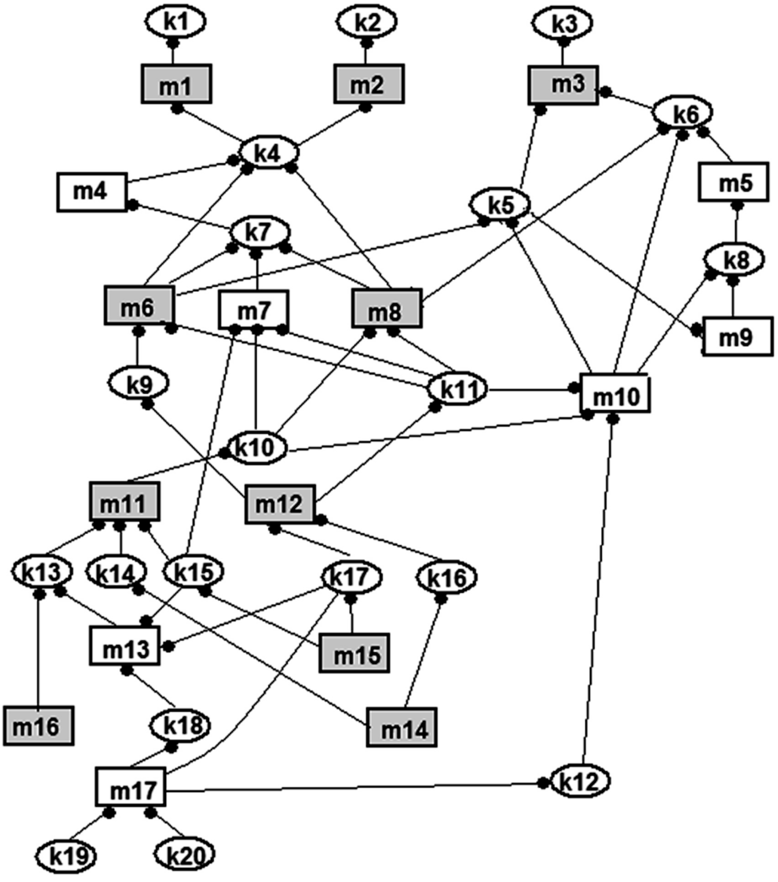


Fig. 1. Educational space model (AND-OR graph *G*)

is formed at the beginning. Front Φ_1^m, including modules which outputs cover set of concepts Φ_0^k is formed at the first step. Set of modules Φ_1^m is connected with a set of target concepts Φ_0^k according to the communication structure of graph *G*.

Front Φ_2^k, including a set of concepts covering all inputs of modules of set Φ_1^m is formed at the second step. Set of modules Φ_1^m is connected with a set of target concepts Φ_2^k according to the communication structure of graph *G*. Further, at each odd step (i = 3, 5, 7, …), front Φmi, including modules which outputs cover set of concepts of front Φ_{i-1}^k, is formed. At each even step (*i* = 4, 6, 8, …), front Φ_i^k, including concepts which outputs cover inputs of set of modules of front Φ_{i-1}^m is formed. The succession of formed fronts is the following: Φ_0^k, Φ_1^m, Φ_2^k, Φ_3^m, Φ_4^k, Φ_5^m, Φ_6^k,… .

Let's consider the process of building a path by way of example of educational space model from work [4].

At first front Φ_0^k is formed including initial set of target concepts $\Phi_0^k = \{k_1, k_2, k_3\}$.

Front Φ_1^m, including modules which outputs cover set of concepts Φ_0^k is formed at the first step. $\Phi_1^m = \{m_1, m_2, m_3\}$. m1 is connected with k_1, m_2 – with k_2, m_3 – with k_3.

Front Φ_2^k, including concepts which outputs cover set of inputs of modules Φ_1^m is formed at the second step. $\Phi_2^k = \{k_4, k_5, k_6\}$. k_4 is connected with m_1 and m_2, k_5 and k_6 – with m_3.

Front Φ_3^m including modules which outputs cover set of concepts Φ_2^k is formed at the third step. $\Phi_3^m = \{m_4, m_6, m_5\}$. m_4 is connected with k_4, m_6 – with k_5, m_5 – with k_6.

Front Φ_4^k including concepts which outputs cover set of modules Φ_3^m is formed at the fourth step. $\Phi_4^k = \{k_7, k_9, k_{11}, k_8\}$. k_7 is connected with m_4, k_9 and k_{11} – with m_6, k_8 – m_5.

Front Φ_5^m, including modules which outputs cover set of concepts Φ_4^k is formed at the fifth step. $\Phi_5^m = \{m_6, m_{12}, m_9\}$. m_6 is connected with k_7, m_{12} – with k_9 and k_{11}, m_9 – with k_8.

Front Φ_6^k including concepts which outputs cover set of modules Φ_3^m is formed at the sixth step. Note that module m6 is already included in Φ_3^m and its outputs are covered, that's why we exclude m6 from Φ_5^m. $\Phi_6^k = \{k_{16}, k_{17}\}$. k_{16}, and k_{17} are connected with m_{12}.

Front Φ_7^m including modules which outputs cover set of concepts Φ_6^k is formed at the seventh step. $\Phi_7^m = \{m_{14}, m_{15}\}$. m_{14} is connected with k_{16}, m_{15} – with k_{17}.

Since nothing comes to inputs of modules Φ_7^m, path forming is completed.

The peculiarity of connecting elements of two adjacent fronts Φ_{i-1}^k and Φ_i^m is that according to initial graph G the same concept can be an output for several modules. A problem of selecting only one connection between concept k_j from front Φ_{i-1}^k and one of modules of front Φ_i^m appears. This explains multitude of learning paths. Connection structure between elements of two fronts Φ_{i-1}^m and Φ_i^k remains unchanged according to connection structure of graph G. The process continues until all module outputs have no other concepts than initial or no concepts at all.

Thus building of specific learning path is confined to selection of alternative connection variants between the fronts during the wave propagation.

Synthesized subgraph of AND-OR graph is shown in Fig. 2 with selected variants of connection between the fronts, with a path from initial concepts (or concepts without outputs) to each target concepts (learning path). Concepts and modules included in the sought-for paths shall be explained in the textbook. The concepts are explained in the modules. To study concept k_i, the learning path shall have at least one module explaining the concept. On the other hand, trainee shall know or preliminary study concepts which are inputs for module m_{ij} to understand module m_j with explanation of concept k_i (let it be module m_{ij}). It is obvious that in general the problem has many solutions. It is necessary to select one optimal among the solutions according to one of criteria which can be presented by some metadata function of the modules included in the path, e.g. complexity of mastering, material modernity, extent of interaction with modules of other subject areas, etc. For example, if user is interested in the briefest textbook and the textbook length is measured according to the number of included modules.

A swarm algorithm for synthesis of optimal learning path, based on ideas of adaptive behavior of ant colony [6, 7] is used in the work. Unlike conventional

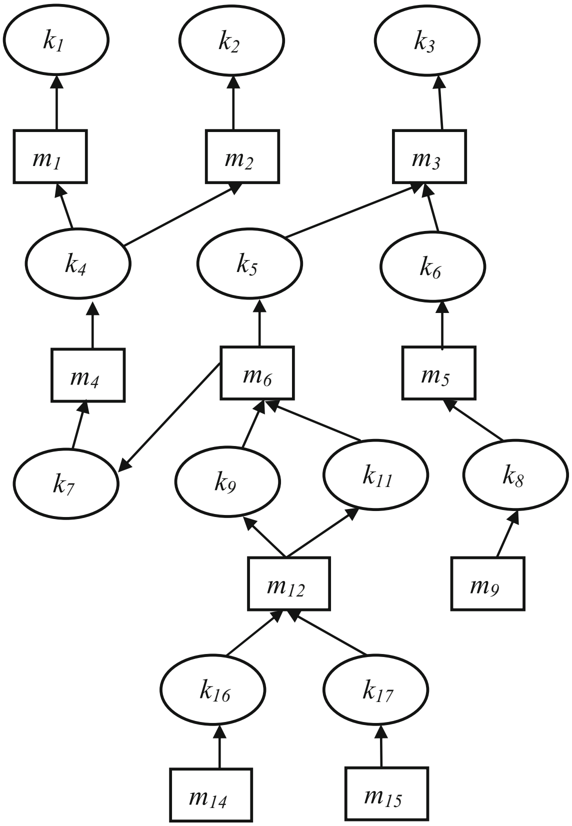


Fig. 2. Built learning path

paradigm of ant algorithm, which is solved with the help of minimum cost path in finding solutions graph, the new swarm algorithm is solved with the help of a subgraph with a dynamic structure, selected in finding solutions graph. ***Stigmergy, a mechanism of indirect coordination***, is the main mechanism in the new paradigm of swarm intelligence and ant colony algorithm.

Presentation of optimization task in the form of swarm intelligence paradigm is based on two key points: creation of finding solutions graph (FSG) in the form of *educational space graph* (AND-OR graph G.) and building of permissible alternative learning paths by a swarm of agents on the finding solutions graph with the help of the wave procedure described above.

Let's form set W_{ϕ} of all lines of graph G, included into all vertexes-modules and set W_a of all lines of graph G, included into all vertexes-concepts. Let's call W_{ϕ} a set of fixed connections, set W_a – a set of alternative connections. Let's sort out subgraph $G^* = (M \cup K, W_a)$, including individual learning path, $G^* \subset G$, into G.

In the general case agent collective perform the task solution $A = \{a_c | c = 1,2,\ldots,n_c\}$. After FSG G is built on all lines of set W_a, initial quantity of pheromone Q/v, where $v = |U|$, is left. Sets of target and initial concepts are formed as initial data. Process of finding solutions is iterative. Each iteration l includes three stages. At the first stage of each iteration of swarm algorithm, each agent ac forms learning path (subgraph) S^*_c = *(MK,W$_{ac}$)* on *educational space model*, where: W_{ac} – set of connections selected by agent a_c, $W_{ac} \subset W_a$; *MK* – set of vertexes of graph G, incident with line W_{ac}. At the second stage each agent ac deposits pheromone on lines W_{ac} of graph G, included in learning path $S_c \subset G^*$, built by the agent. At the third stage the pheromone vapors on lines W_a of graph G. Ant-cycle method is used in the work. In this case pheromone is deposited by agents on lines W_a of graph G^* after the solution is completed.

Step-by-step process of learning path creation is made by propagating a wave to educational space model starting from starting front Φ^k_0. At step i of wave propagation an agent forms front Φ_i and establishes connections between Φ_i and front Φ_{i-1}. Two cases are possible.

In the first case at step i front of concepts Φ^k_i is formed, at step *i-1* front of modules Φ^m_{i-1} is formed. In this case connection structure between Φ^m_{i-1} and Φ^k_i remains unchanged, i.e. the same as in graph G, since inputs of modules from front Φ^m_{i-1} is a connective AND.

In the second case at step i front of modules Φ^m_i is formed, and at step *i-1* front of concepts Φ^k_{i-1} is formed. Since each vertex k_n (concept) of graph G is a connective OR, i.e. all inputs correspond t the same concept, then one connection is left to avoid doubling at input of vertex k_n. The agent selects a connection for each concept. Let's assume that R_n is a set of lines u_{nm} included into vertex $k_n \in \Phi^k_{i-1}$. For each line $u_{nm} \in R_n$ parameter f_{nm}, total level of pheromone at this line, is determined. Probability P_{nm} of selecting line u_{nm} is determined by the following formula

$$P_{nm} = f_{nm} / \sum\nolimits_n (f_{nm}) \tag{1}$$

Agent with probability P_{nm} selects one of lines, which remains in the connection structure of learning path being built S_c between Φ^k_{i-1} and Φ^m_i. Connections are selected until all inputs of concepts of front Φ^k_{i-1} (vertex of graph G) have only one connection.

Set $W_{ac} \subset W_a$ of all lines, included into vertexes-concepts, is sorted out in the built path S_c.

At the second stage of iteration, each agent a_c deposits pheromone on lines of graph G, belonging to a set $W_{ac} \subset W_a$.

Amount of pheromone $\tau_c(l)$, deposited by agent a_c at each line $u_{nm} \in W_{ak}$ of graph G, is proportional to basic (pivotal) amount of pheromone - Δ and is determined as follows

$$T_c(l) = \Delta / F_c(l), \tag{2}$$

where l – iteration number, $F_c(l)$ – target function for solution, corresponding to learning path S_c. Usually a number of modules included into S_c. is considered as $F_c(l)$. The less $F_c(l)$, the more pheromone is deposited in graph G on lines of set W_{ac} and, consequently, the higher is the probability of selecting these lines for building the learning path at the next iteration.

After each agent formed solution (learning path) S_c and deposited pheromone, at the third stage pheromone vapors on lines of set W_{a} of graph G according to formula

$$f_{nm} = f_{nm}(1 - \rho), \tag{3}$$

where ρ is a coefficient of renewal. After all actions are completed, the agent with the best solution is in iteration and the best solution is stored. Then transition to the next iteration is made.

Metaheuristic of ant algorithm is based on combination of two techniques: common pattern is built on basic method which includes one or another integrated procedure. An important aspect is that the integrated procedure is in most cases an independent algorithm for solution of the same task as the metaheuristic method as a whole. The basic method consists in realization of iterative search procedure of the best solution on the basis of mechanisms of adaptive behavior of ant colony. The integrated procedure is based on constructive algorithm of building some specific solution interpretation by an ant. Constructive unit, activity of artificial ants, plays a key role in optimization with the help of ant colonies [8].

2.1 Agent Swarm Behavior Algorithm

1. Educational space model (AND-OR graph *G*) $G = (M \cup K, U)$ is created, where M – set of modules, K – set of concepts.
2. Set of target $K_t \subset K$ and initial $K_i \subset K$ concepts is selected.
3. The following is assumed: initial quantity of pheromone – Q; ρ – coefficient of pheromone renewal on lines of graph G; V – number of agents in a swarm; L – number of iterations.
4. Initial amount of pheromone Q/v, where $v = |\, W_a \,|$ is deposited on a set of lines $Wa \subset U$ of graph G, included into vertexes-concepts $kj \in K$.
5. Front Φ_{k0} including a set of target concepts is formed.
6. l = 1. (l – iteration number).
7. c = 1. (c – agent number).
8. “Agent algorithm” is performed.
9. If c < V, then c = c + 1 and transition to item 8, otherwise transition to item 10.
10. Each agent ac deposits pheromone in subgraph G on lines of set $Wac \subset Wa$ in learning path Sc, built by it in the following amount

$$\tau_c(l) = \Delta / F_c(l) \tag{4}$$

Total amount of pheromone $h_i(l)$, deposited on line $u_i \in W_{ac}$ after completion of l iterations, is determined as

$$h_i(l) = h(l-1) + \sum_{c|u_i \in W_{ac}(l)} \tau_c(l) \tag{5}$$

11. After each agent deposited pheromone, the pheromone vapors on lines of set W_a of graph G according to the following formula.

$$h_i = h_i(1 - \rho) \tag{6}$$

where ρ – coefficient of renewal, h_i – total amount of pheromone, deposited by ants on line $u_i \in W_a$ of graph G.
12. Selection of the best solution received during all performed iterations.
13. If all iterations are performed, then algorithm operation ends, otherwise $l = l + 1$ and transition to item 8.

2.2 Agent Algorithm

1. $i = 1$. (i – number of wave propagation step).
2. Front Φ^{k}_{i-1} is included into learning path S_c.
3. Front of modules Φ^{m}_{i} connected in graph G with concepts of front Φ^{k}_{i-1} is formed. Front of modules Φ^{m}_{i} in included into learning path S_c.
4. Successive vertex-concept $k_n \in \Phi^{k}_{i-1}$ is selected beginning with the first one.
5. Set R_{n} of lines u_{nm} included into vertex $k_n \in \Phi^{k}_{i-1}$ 1 is determined on graph G.
6. For each line $u_{nm} \in R_{\mathrm{n}}$ parameter f_{nm}, total level of pheromone at this line, is determined in graph G.
7. Probability P_{nm} of selection is calculated for each line $u_{nm} \in R_n$

$$P_{nm} = f_{nm} / \sum n(f_{nm}) \tag{7}$$

8. Agent ac with probability P_{nm} selects one of lines $u_{nm} \in R_m$, which remains in the connection structure of learning path being built S_{k} between Φ^{k}_{i-1} and Φ^{m}_{i}.
9. If all vertexes $k_n \in \Phi^{k}_{i-1}$ are viewed, then transition to item 10, otherwise transition to item 4.
10. If nothing comes to inputs of modules Φ^{m}_{i}, then transition to item 15, otherwise transition to item 11.
11. $i = i + 1$.
12. Front of vertexes-concepts Φ^{k}_{i} with connections in graph G with vertexes of front Φ^{m}_{i-1} is formed. Front of concepts Φ^{k}_{i} is included into learning path S_c.
13. Agent ac records all connections between fronts Φ^{m}_{i-1} and Φ^{k}_{i}, included into graph G, in learning path S_c.
14. $i = i + 1$. Transition to item 3.
15. Evaluation $F_c(l)$ of learning path S_c, built by agent *ac* at iteration l is calculated.
16. The algorithm operation ends.

This algorithm operation time depends on colony l (number of iterations) life cycle, number of vertexes n of graph G, number of agents m, and is defined as (l•n•m).

3 Experimental Research

The algorithm for the optimal learning paths Formation was implemented by means of C# for Windows platform. Moreover, all the experiments were carried out on the following computer: Intel® Core ™ i5 CPU 3.33 GHz, 4 GB RAM. There were 2 objectives to complete experiments: effectiveness and ant colony mechanisms quality research to solve the problem for synthesis of optimal learning path. For these purposes, the synthesis procedure with known optimum was carried out on test data.

The first objective was to study the impact of control statements, such as: ant population size m, number of iterations l, and parameters that control the deposition and evaporation of the pheromone.

During the study of the algorithm convergence for each experiment the number of generation was saved, after which no better assessment was gained. In each series of 50 experiments minimum and maximum generation number were determined. Also the average number of generations with no better assessment was calculated. The conducted experiments showed that the algorithm converges after 120 iterations when a population volume $V = 100$. The first objective was to study the impact of control statements, such as: ant population size m, number of iterations l, initial pheromone quantity Q deposited on graph edges.

Experiments have shown that the initial amount of pheromone Q must be 10–12 times greater than the average amount of pheromone $\tau_c\ (l)$, deposited at each iteration.

Comparison of test values obtained by ant algorithm run on test data with known optimum, showed that 90% of the examples show the best solution, 2% examples of solutions were 3% worse than optimal, and in 3% of examples of decisions were worse than no more than 1%.

The time complexity of the algorithm, obtained by experiments is in the range of $O(n^2) - O(n^3)$.

4 Conclusion

The proposed algorithm allows generating variants of learning paths consisting of subsets of small modules (divided content units) of educational materials. Method of learning path optimization based on ontological approach is considered. A swarm algorithm for synthesis of optimal learning path, based on models of adaptive behavior of ant colony is proposed. The distinctive feature is graphic presentation of learning path convenient for analysis of the obtained results. Experimental studies were performed on IBM PC. The experiments have showed that the best solution results for graphs with 1000 maximum were obtained in average for 120 iterations by a swarm from 100 agents. Probability of obtaining an optimal solution is 0.9. Time complexity (TC) for the considered test tasks is $(TC \approx O(n^2))$, where n is a number of graphs G.

Acknowledgements. This research is supported by grant of the Russian Science Foundation (project # 14-11-00242) in the Southern Federal University.

References

1. Bova, V.V.: Conceptual model of knowledge representation when building intelligent information systems. Izvestiya of SFedU. Eng. Sci. **7**(120), 109–117 (2014)
2. Norenkov, I.P.: Ontological methods of electronic textbook synthesis. J. Res. Pract. “Otkrytoe Obrazovanie” **6**, 39–44 (2010)
3. Kravchenko, Yu.A., Markov, V.V.: Ontological approach for building information resources on the basis of diverse knowledge sources. Izvestiya of SFedU Eng. Sci. **7**(144), 116–120 (2013)
4. Norenkov, I.P., Sokolov, N.K.: Synthesis of individual learning paths in ontological teaching systems. Informatsionnie Tekhnologii **3**, 74–77 (2009)
5. Telnov, Yu.F.: Intelligent Information Systems, p. 26. Moscow State University of Economics, Statistics and Informatics, Moscow (2003)
6. Lebedev, O.B.: Models of Adaptive Behavior of Ant Colony in Design Tasks. SFedU Publishing House, Taganrog (2013)
7. Lebedev, B.K., Lebedev, V.B.: Optimization by crystallisation of alternatives deposit. Izvestiya SFedU **7**, 11–17 (2013). TTI SFedU Publishing House
8. Lebedev, B.K., Lebedev, O.B.: Modelling of adaptive behavior of ant colony when searching solutions interpreted by trees. Izvestiya SFedU **7**, 27–35 (2012). TTI SFedU Publishing House

Empirical Evaluation of the Cycle Reservoir with Regular Jumps for Time Series Forecasting: A Comparison Study

Mais Haj Qasem[1](✉), Hossam Faris[1], Ali Rodan[1], and Alaa Sheta[2]

[1] King Abdullah II School for Information Technology,
The University of Jordan, Amman, Jordan
mais_hajqasem@hotmail.com

[2] Computing Science Department, Texas A&M University-Corpus Christi,
Corpus Christi, TX, USA

Abstract. The cycle reservoir with regular jumps (CRJ) is a recent deterministic reservoir model with a very simple structure and highly constrained weight values. CRJ was proposed as an alternative to the randomized Echo State Network (ESN) reservoir. In this work, we empirically evaluate the performance of CRJ for time series forecasting problems, and compare it to ESN and Auto-Regressive with eXogenous inputs (NARX) models. The comparison is conducted based on seven time series datasets that represent different real world cases. Simulation results show that CRJ outperforms ESN and NARX models. The results also demonstrate the effectiveness of CRJ when applied for different time series forecasting problems

Keywords: Echo State Network · Recurrent neural network · Reservoir Computing · Times series data · Forecasting

1 Introduction

Time series data are observations of well-defined data items that represent repeated measurements over a period of time, such as a month, quarter, or year [1]. A time series shows the actual movements in the data over time caused by cyclical, seasonal, and irregular events on the data item being measured. Time series data are used in different areas such as statistics, signal processing, pattern recognition, earthquake prediction, weather forecasting, trajectory forecasting, control engineering and communications engineering.

The development of a time series forecasting for nonlinear behavior represents a challenge for both engineers and mathematicians. Typically, nonlinear time series modeling involves two major steps: the selection of a model structure with a certain set of parameters and the selection of an algorithm which estimate these parameters. The later issue usually biases the former one. There are still many unsolved problems related to the implementation and design of time series models.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_12

In literature, there is a wide range of machine learning algorithms proposed and applied for the task of time series forecasting. One of the common types of these algorithms is the Echo State Networks (ESN) [2]. ESN is a supervised learning recurrent neural network (RNN) with a fixed random hidden (reservoir) layer and a memoryless readout. The aim of ESN is to drive large, random and fixed RNN for the input signal, where the neurons (units) are promoted in the reservoir network through a nonlinear response signal [3]. The desired output signal is merged with a trainable linear combination of all response signals. The basic idea of ESN is similar to that of the liquid state machine (LSM), which was developed independently by Mass et al. [4].

However, standard ESN possesses several drawbacks, which affect its acceptability. First, the fixed random reservoir is difficult to understand. Second, the reservoir specification and input connections require many trials. Third, imposing a constraint on the spectral radius of the reservoir matrix is useless when setting the reservoir parameter [5]. Lastly, the reservoir's connectivity and weight structure are not optimal, and the reservoir's dynamic organization is still unclear.

In attempt to overcome these problems, Rodan and Tino [6] proposed the deterministic Cycle Reservoir with regular Jumps (CRJ). CRJ is considered as a new class of state-space reservoir models where it possesses a fixed state transition structure (the "reservoir") and an adjustable readout from the state space as in all Reservoir Computing (RC) models.

CRJ has highly-constrained weight values while the nodes in the reservoir are connected into a unidirectional cycle with a fixed value r_c, similar to that of the Simple Cycle Reservoir (SCR) [7]. In addition to that, a bidirectional jump weight r_j is found which serves as a shortcut for the CRJ network. Previous works have shown that the addition of these regular jumps can improve the performance of the model [6]. Recently, CRJ has shown very promising performance in different types of applications [8,9].

In this work, we investigate the application of CRJ [6] for different time series forecasting problems. For this purpose, seven time series datasets of different real world applications are utilized. The performance of the developed CRJ model is evaluated and compared with ESN [2,10], and the NARX model. The ultimate goal of this study is to reveal the efficiency of CRJ when used for times series forecasting in different applications.

This paper is organized as follows: all methods used in this work for the task of time series forecasting are described in Sect. 2. The selected time series datasets for the purpose of evaluating and benchmarking are presented in Sect. 3. The details of the conducted experiments and the discussion are given in Sect. 4. Finally, the findings of this work are summarized in Sect. 5.

2 Methods

2.1 Echo State Network (ESN)

ESN is a discrete time recurrent neural network with $\{A\}$ input units, $\{M\}$ internal units and $\{S\}$ output units over discrete time slots $n = \{1, 2, 3, \dots\}$.

The activation of the ESN is expressed using a vector for each layer, as given in Eq. 1.

$$a(n) = a_1(n), \dots a_A(n))^T, b(n) = (b_1(n), \dots b_M(n))^T, c(n) = (c_1(n), \dots c_S(n))^T \tag{1}$$

The linking weights between the neurons are gathered in $M \times A$ size matrix, for the input, which is referred to by $W^{in} = (W_{ji}^{in}), M \times M$ size matrix for the internal weight, which is referred to by $W = (w_{ij}), S \times (A + M)$ size matrix for the output, which is referred to by $W^{out} = w_{ij}^{out}$, and $M \times S$ size matrix for the connection that projects back from output to the internal unit, which is referred to by $w_{back} = w_{ij}^{back}$.

Unlike RNN, where all the weights for the inputs, internals and output are adaptable, in ESN the reservoir connection weights as well as the input weights are randomly generated and fixed (non trainable). The only trainable part is the output weights. The fixed random weights for the input and reservoir layers are then scaled with a chosen values, v for the input and λ for the reservoir, where $v, \lambda \in (0, 1)$. Moreover, the spectral radius of the reservoir matrix should be less than 1 to ensure a sufficient condition for the "echo state property" (ESP). By doing so, ESN ensures that the reservoir state is an "echo" for all input history. The internal units are updated, when moving from time slot n to time slot $n+1$, according to Eq. 2.

$$b(n + 1) = f(W^{in}a(n + 1) + Wb(n) + W^{back}c(n) + z(n + 1)) \tag{2}$$

Where f is the reservoir activation function (usually tangent hyper function (tanh)) and z is optional small white noise that might be needed in some cases for solving the overfitting problem. The linear output is computed using Eq. 3.

$$c(n + 1) = f^{out}(W^{out}x(n + 1)) \tag{3}$$

Where $f_{out} = (f_{out}^1, \dots f_{out}^S)$ are the output units function and $x(n + 1) = [b(n+1); a(n+1)]$ are a concatenation for the internals and the input activation vectors.

2.2 Cycle Reservoir with Regular Jump (CRJ)

CRJ is a simple deterministic reservoir model with highly constrained weight values [6]. CRJ deterministically generates reservoir that could have the potential of a better performance than standard ESN and other models previously proposed in the literature [7].

To implement CRJ, you need to optimize several parameters including the cycle weight r_c, jump weight r_j, input weight v. Then, the reservoir size N, similar to ESN, is determined. Moreover, the number of input and output units with the added bias value to input units are also determined based on the nature of the task and the target output.

Unlike ESN, CRJ has a simple regular topology with full connectivity between the input and reservoir, there is no need to specify different weight value for each connection between two nodes, where all the reservoir nodes connected via unidirectional cycle with the same value r_c. The value of r_c should be on the range of [0,1].

CRJ also has a bidirectional shortcut (jumps) between the reservoir units r_j. These jumps increase the density of the connections in the internal units, which in term facilitates a good training. Unlike ESN, which generates the input weight randomly, CRJ required to set its input weight sign values in a complete deterministic manner. The deterministic input signs are generated based on the π digits where each digit is thresholded at 4; if the value of the digit is between $0 \leqslant digit \leqslant 4$ then the connection sign will be minus $(-)$, and if the value of the digit is between $5 \leqslant digit \leqslant 9$ the connection sign will be positive $(+)$.

2.3 Auto-regressive with eXogenous Inputs (NARX)

NARX model was first presented in 1985 by Leontaritis and Billings [11,12] as a means of describing the input-output relationship of a nonlinear system [13]. Time Series prediction using the NARX model was explored in many articles [14,15]. The general NARX model structure can be represented using the following nonlinear differential equation:

$$y(t) = f(y(t-1), \dots, y(t-n), u(t-1), \dots, u(t-m)) \tag{4}$$

f represents a nonlinear mapping between the system input $u(t)$ and the past outputs $y(t-1), y(t-2), \dots$. The order of the model input and output is assumed to be n and m, respectively. The NARX model can be represented as given in Eq. 4. The model parameters can be estimated using LSE.

$$\begin{aligned} y(t) = a_0 &+ \sum_{k_1=1}^{n} f_{k_1}(x_{k_1}) \\ &+ \sum_{k_1=1}^{n} \sum_{k_2=k_1}^{n} f_{k_1 k_2}(x_{k_1}(t), x_{k_2}(t)) + \dots \\ &+ \sum_{k_1=1}^{n} \cdots \sum_{k_l=k_{l-1}}^{n} f_{k_1 k_2 \dots k_l}(x_{k_1}(t), \dots, x_{k_l}(t)) \end{aligned} \tag{5}$$

Given that:

$$\sum_{k_1=1}^{n} \cdots \sum_{k_l=k_{l-1}}^{n} f_{k_1 k_2 \dots k_z}(x_{k_1}(t), \dots, x_{k_z}(t)) = a_{k_1 k_2 \dots k_z} \prod_{i=1}^{z} x_{k_i}(t) \tag{6}$$

z is in the interval of $[1, l]$. $a_{k_1 k_2 \dots k_z}$ are the model parameters to be estimated.

$$x_k(t) = \begin{cases} y(t-k) & \text{if}\, 1 \leq k \leq n \\ u(t-(k-n)) & \text{if}\, n+1 \leq k \leq n+m \end{cases}$$

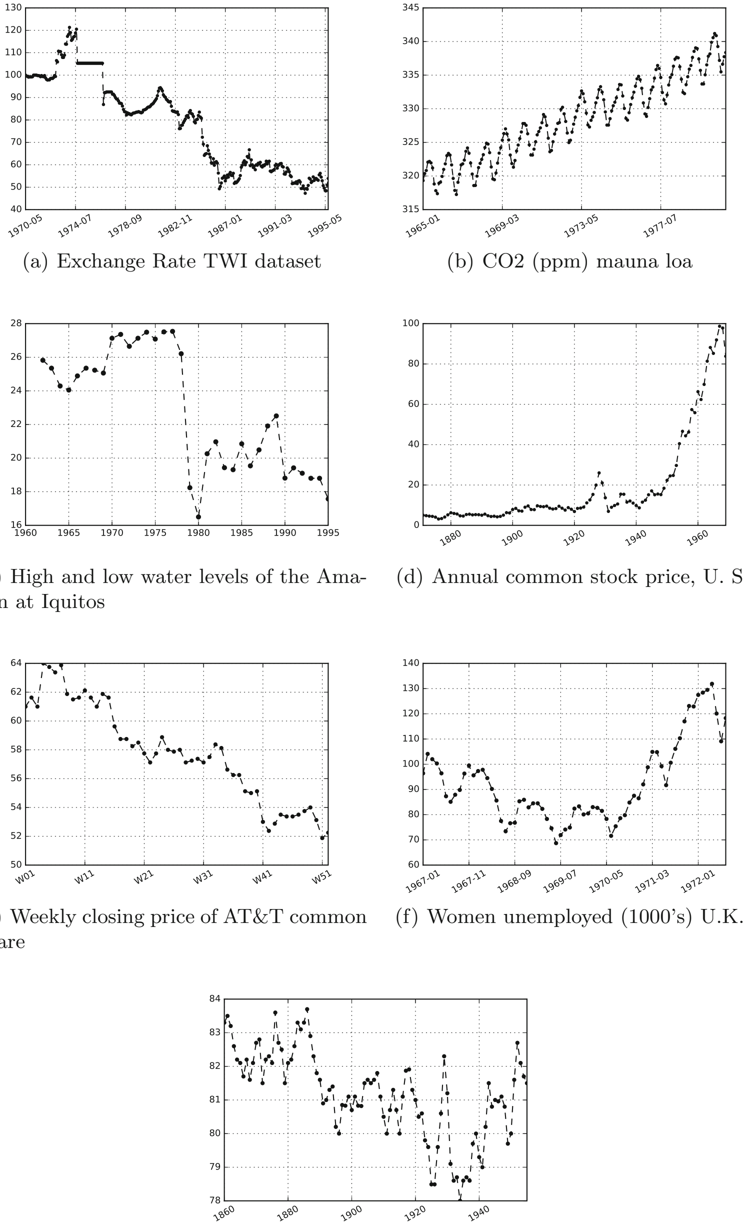
Identifying a NARX model requires two steps: (1) pick the best model structure [16], (2) estimating the model parameters. NARX model was used to solve many time series analysis, modeling and identification of nonlinear systems [17,18].

3 Datasets

Seven time-series datasets are drawn from DataMarket Repository for experimenting and benchmarking the described models in the previous section. The DataMarket Repository is sponsored by Qliktech. DataMarket delivers intuitive platform solutions for self-service data visualization, guided analytic applications, embedded analytics, and reporting to approximately 40,000 customers worldwide [19]. The selected datasets for our experiments represent real world data including: financial forecasting problems, unemployment rates, environmental modeling and pollutants concentrations. The datasets describe a variety of problems over different time periods and have different levels of complexity. All datasets are described in Table 1 and depicted in Fig. 1.

Table 1. Dataset

	Dataset name	Feature
1	Exchange Rate TWI	304 fact values in 1 monthly time series
2	CO2 (ppm) mauna loa	192 fact values in 1 yearly time series
3	High and low water levels of the Amazon at Iquitos	34 fact values in 1 yearly time series
4	Annual common stock price, U.S.	112 fact values in 1 yearly time series
5	Weekly closing price of AT&T common share	56 fact values in 1 Weekly time series
6	Women unemployed (1000's) U.K.	80 fact values in 1 monthly time series
7	Highest mean monthly level, Lake Michigan	109 fact values in 1 monthly time series



(a) Exchange Rate TWI dataset

(b) CO2 (ppm) mauna loa

(c) High and low water levels of the Amazon at Iquitos

(d) Annual common stock price, U. S

(e) Weekly closing price of AT&T common share

(f) Women unemployed (1000's) U.K.

(g) Highest mean monthly level, Lake Michigan

Fig. 1. Seven representative time series.

4 Experiments and Result

4.1 Parameters Settings

All datasets are divided into two sets for training and testing; 70% was used for training, and the rest is used for testing. In order to obtain the best performance of each model in terms of lowest Error, the parameters of the models should be optimized. Therefore, multiple values for each parameter are tested, and the best value is used to obtain the final output.

- **Echo State Network (ESN):** For spectral radius (λ), 20 different values in the range [0.05–1] were tested. For connectivity (con), 10 different values in the range [0.05–0.5] were tested. The model was also tested under different internal unit sizes (N) in the range [50–500]. The final and best parameter's values for the seven datasets used are shown in Table 2.
- **Cycle Reservoir with Regular Jump (CRJ):** For internal unit weights (r_c) and (rj), 20 different values for each parameter in the range [0.05–1] were tested. For input unit scaler (v), 20 different values in the range [0.05–1] were tested. Also as in ESN, the model was also tested under different internal unit sizes (N) in the range [50–500]. For the jump size, $N/2$ different jump values were tested. The final and best parameter's values for the seven datasets are provided in Table 3.
- **NARX:** The Levenberg-Marquardt training algorithm is utilized to train the model for all the datasets. Different number of hidden nodes are tested for each dataset starting with 5 neurons up to 50 with a step of 5. The results are reported for this model by averaging the obtained RMSE values over 10 independent runs. Evaluation results along with the best parameters are shown in Table 4 for the seven datasets.

Evaluation of the models performance will be done via Normalized Mean Square Error (NMSE), as given in Eq. 7.

$$RMSE = \sqrt{\frac{\sum_{n=1}^{T}(\hat{c}(n) - c(n))^2}{T}} \tag{7}$$

Where $\hat{c}(n)$ is a predicted output, $c(n)$ is a desired output.

4.2 Comparison Results

The final evaluation results of the CRJ, ESN and NARX models are summarized and listed in Table 5. The results are presented in terms of RMSE and the standard deviation of the 10 independent runs of each model (denoted as RMSE ± STD). Note that CRJ models don't have a standard deviation since they are deterministic models. On the other had, ESN and NARX yielded different RMSE results over different datasets due to their random weight generation.

As it can be noticed in the evaluation results, the CRJ model showed the lowest RMSE values for all datasets. Examining the results of the other two models, we can't see a dominant model between NARX and ESN as a second best model for all datasets. ESN model achieved the second best evaluation results in four datasets (3, 4, 5, 6), while NARIX model was the second best in three datasets (1, 2, 7). Moreover, comparing NARIX to ESN in terms of standard deviation, we can notice that NARX model is more robust since it showed lower values in most of datasets, while ESN showed noticeably high values of standard deviation in Datasets 4 and 6. Overall, we can conclude that the CRJ model is very efficient when applied for complex time series forecasting problems.

Table 2. ESN result

Dataset	λ	con	N	RMSE
1	0.05	0.4	125	2.83
2	0.1	0.35	325	2.57
3	0.95	0.2	50	1.80
4	0.05	0.05	175	1.40
5	0.95	0.35	150	1.48
6	0.8	0.05	300	2.02
7	0.05	0.15	150	2.57

Table 3. CRJ result

Dataset	r_c	v	r_j	Step size	N	RMSE
1	0.05	0.3	0.05	1	450	1.72
2	0.1	0.05	0.1	10	100	1.23
3	0.95	0.6	0.05	20	235	1.38
4	0.05	0.05	0.6	1	50	1.02
5	0.05	0.1	0.05	1	400	1.25
6	0.1	0.95	0.05	10	75	1.05
7	0.05	0.2	0.05	1	500	0.84

Table 4. NARX result

Dataset	N	Delay	RMSE
1	10	2	2.59
2	50	2	1.99
3	10	2	5.71
4	20	2	2.79
5	10	2	1.56
6	10	2	4.70
7	50	2	1.45

Table 5. Comparison result.

Dataset	ESN	CRJ	NARIX
	RMSE ± STD	RMSE	RMSE ± STD
1	2.83 ± 1.81	1.72	2.59 ± 1.53
2	2.57 ± 2.22	1.23	1.99 ± 2.25
3	1.80 ± 2.68	1.38	5.71 ± 1.86
4	1.40 ± 4.41	1.02	2.79 ± 1.94
5	1.48 ± 1.53	1.25	1.56 ± 1.60
6	2.02 ± 5.76	1.05	4.70 ± 1.49
7	2.57 ± 2.39	0.84	1.45 ± 1.32

The CRJ model has the advantage of simplicity and robustness when compared to other well known models like ESN and NARX.

5 Conclusion

In this work, the application of cycle reservoir with jumps (CRJ) model is evaluated for time series forecasting. For the purpose of benchmarking and evaluation, seven time series datasets that describe different real world applications are utilized. The evaluation results of CRJ are compared with those obtained for two well regarded models which are the Echo State Network (ESN) and Auto-Regressive with eXogenous inputs (NARX). Evaluation results showed that CRJ achieved the lowest RMSE in all datasets. Subsequently, we argue that CRJ has the potential to outperforms ESN and NARIX in terms of accuracy and robustness when applied to time series forecasting problems.

References

1. Thakur, G.S., Thakur, R.S., Thakur, R.S.: Design of 2-level clustering framework for time series data sets. In: Deep, K., Nagar, A., Pant, M., Bansal, J. (eds.) Proceedings of the International Conference on Soft Computing for Problem Solving (SocProS 2011). Advances in Intelligent and Soft Computing, vol. 131, pp. 205–212. Springer, New Delhi (2012)
2. Jaeger, H.: The "echo state" approach to analysing and training recurrent neural networks-with an Erratum note. Bonn, Ger.: Ger. Natl. Res. Cent. Inf. Technol. GMD Tech. Rep. **148**(34), 13 (2001)
3. Jaeger, H., Lukoševičius, M., Popovici, D., Siewert, U.: Optimization and applications of echo state networks with leaky-integrator neurons. Neural Netw. **20**(3), 335–352 (2007)
4. Maass, W., Natschläger, T., Markram, H.: Real-time computing without stable states: a new framework for neural computation based on perturbations. Neural Comput. **14**(11), 2531–2560 (2002)

5. Schrauwen, B., Defour, J., Verstraeten, D., Campenhout, J.: The introduction of time-scales in reservoir computing, applied to isolated digits recognition. In: Sá, J.M., Alexandre, L.A., Duch, W., Mandic, D. (eds.) ICANN 2007. LNCS, vol. 4668, pp. 471–479. Springer, Heidelberg (2007). doi:10.1007/978-3-540-74690-4_48
6. Rodan, A., Tiňo, P.: Simple deterministically constructed cycle reservoirs with regular jumps. Neural Comput. **24**(7), 1822–1852 (2012)
7. Rodan, A., Tino, P.: Minimum complexity echo state network. IEEE Trans. Neural Netw. **22**(1), 131–144 (2011)
8. Qasem, M.H., Al Assaf, M.M., Rodan, A.: Data mining approach for commercial data classification and migration in hybrid storage systems. World Acad. Sci. Eng. Technol. Int. J. Comput. Electr. Autom. Control Inform. Eng. **10**(3), 481–484 (2016)
9. Rodan, A., Faris, H.: Credit risk evaluation using cycle reservoir neural networks with support vector machines readout. In: Nguyen, N.T., Trawiński, B., Fujita, H., Hong, T.-P. (eds.) ACIIDS 2016. LNCS (LNAI), vol. 9621, pp. 595–604. Springer, Heidelberg (2016). doi:10.1007/978-3-662-49381-6_57
10. Jaeger, H.: Adaptive nonlinear system identification with echo state networks. Networks **8**(9), 17 (2003)
11. Leontaritis, I.J., Billings, S.A.: Input-output parametric models for non-linear systems. Part I: deterministic non-linear systems. Int. J. Control **41**(2), 303–328 (1985)
12. Leontaritis, I.J., Billings, S.A.: Input-output parametric models for non-linear systems. Part II: stochastic non-linear systems. Int. J. Control **41**(2), 329–344 (1985)
13. Mohamed Vall, O.M., M'hiri, R.: An approach to polynomial NARX/NARMAX systems identification in a closed-loop with variable structure control. Int. J. Autom. Comput. **5**(3), 313–318 (2008)
14. Menezes Jr., J.M.P., Barreto, G.A.: Long-term time series prediction with the narx network: an empirical evaluation. Neurocomputing **71**(16–18), 3335–3343 (2008)
15. Pisoni, E., Farina, M., Carnevale, C., Piroddi, L.: Forecasting peak air pollution levels using NARX models. Eng. Appl. Artif. Intell. **22**(4–5), 593–602 (2009)
16. Ho, C.K.S., French, I.G., Cox, C.S., Fletcher, I.: Genetic algorithms in structure identification for NARX models. In: Smith, G.D., Steele, N.C., Albrecht, R.F. (eds.) Artificial Neural Nets and Genetic Algorithms, pp. 597–600. Springer, Vienna (1998)
17. Menezes Jr, J.M.P., Barreto, G.A.: A new look at nonlinear time series prediction with NARX recurrent neural network. In: The 2006 Ninth Brazilian Symposium on Neural Networks, pp. 160–165, October 2006
18. Diaconescu, E.: Prediction of chaotic time series with NARX recurrent dynamic neural networks. In: Proceedings of the 9th WSEAS International Conference on International Conference on Automation and Information. ICAI 2008, pp. 248–253. World Scientific and Engineering Academy and Society (WSEAS) (2008)
19. Qliktech: DataMarket Repository (2017). Accessed 12 Feb 2017

Forecasting of Convective Precipitation Through NWP Models and Algorithm of Storms Prediction

David Šaur(✉)

Faculty of Applied Informatics, Tomas Bata University in Zlin,
Nad Stranemi 4511, Zlin, Czech Republic
saur@fai.utb.cz

Abstract. This article focuses on contemporary possibilities of forecasting of convective storms which may cause flash floods. The first chapters are presented predictive tools such as numerical weather prediction models (NWP models) and the algorithm of convective storms prediction, which includes a storm prediction based on the principles of mathematical statistics, probability theory and artificial intelligence methods. Discussion section provides outputs from the success rate of these forecasting tools on the historical weather situation for the year 2016. The Algorithm's output may be useful for early warning of population and notification of crisis management authorities before a potential threat of flash floods in the Zlin Region.

Keywords: Weather forecast · Convective precipitation · Flash floods · Crisis management · Early warning · Artificial intelligence

1 Introduction

Incidence of storm situation that caused the flash floods, have been steadily increasing every year. In the past, floods were induced by extensive and persistent rain, especially in the years 1997, 2002 and 2006 [1]. Flash floods as a phenomenon of our time have begun to regularly occur since 2007. Although this type of flood did not cause excessive damage (in the order of several billion Czech crowns) as the first type of flood (hundreds of billions of Czech crowns), just the high frequency of occurrence has been the main impulse to the improvement of early warning of population and preventive measures against the flash floods [2].

Flash floods are caused by meteorological and hydrological factors, particularly high intensity rainfall, slow and the stationary motion of precipitation and high soil saturation [3, 4]. Meteorological factors are predicted by NWP models [5–7], nowcasting and expert meteorological systems [8–10]. Except these forecasting systems, prediction of flash flood and heavy rainfall has been also investigated by methods of artificial intelligence in the neural networks [11, 12], especially Backpropagation algorithm [13]. Hydrological factor of soil saturation is published through a Flash Flood Guidance from the Czech Hydrometeorological Institute [4]. Combination of hydrometeorological factors was investigated and implemented by the Algorithms of

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_13

storm prediction. This algorithm has been experimentally developed in the author's dissertation. The Algorithm storms prediction is a forecasting tool the occurrence of convective precipitation and other dangerous accompanying phenomena (hail, strong wind gusts and tornadoes) induced by convective storms including forecasts of the risk of flash floods. The primary input data are data from NWP models provided in image formats PNG and JPEG. These image formats are converted with the selected image processing services to the coefficients of hydrometeorological factors specified under the proposed classification of algorithm. Secondary input data is data from historical weather situations. These data are classified by selected meteorological attributes such as temperature, humidity and wind at different levels of the atmosphere. The forecast of convective of precipitation is calculated by Backpropagation algorithm.

The main objective of this article is to compare the success rate of three methods of forecast storms for the purpose of deployment of the most successful prediction tools in experimental mode for a distribution forecast and warnings of crisis management of the Zlin Region.

2 Forecasting of Convective Storms

At present, forecasting of convective storms and their impacts is very complicated. In addition, current predictive possibilities are limited by the size of the predicted surface area (only for the regions and districts). The forecast of convective storms is focused on their symptoms, especially the interaction of significant hydrometeorological factors. The major forecasting tools are:

1. NWP models.
2. Algorithm of convective storms.
3. Prediction based on mathematical statistics and artificial intelligence.

2.1 Numerical Weather Prediction Models

Numerical weather prediction model (NWP model) is a forecasting model designated to weather forecasting. NWP models consist of a dynamic core, a set of parameterization, a model of the earth's surface and assimilated input data from ground-based meteorological stations and radiosondes [14].

NWP models contain a large number of mathematical equations describing the physical phenomena in the atmosphere. Typical ones are the equations of motion, heat exchange, parameterizations for solar radiation, continuity equation, balance equation of water vapour and the laws of energy conservation, etc. However, one of the main equations is the tools of hydrostatic equilibrium, which expresses the dependence of air pressure ρ of the vertical coordinate z:

$$\frac{\partial \rho}{\partial z} = -\mathrm{g}\rho, \tag{1}$$

where g is the size of the gravity acceleration and ρ is the air density. The essence of this equation is the premise of an existing of the balance between the vertical force component of the pressure gradient and the force of gravity. All NWP models are based on the principle of hydrostatic equilibrium. This assumption can be used for modelling of persistent precipitation (rain, snow). Nevertheless, in reality the earth's atmosphere is compressible and hydrostatic balance can be disrupted by convection air updraft [15].

In practice, we distinguish two types of models of hydrostatic equilibrium:

- Hydrostatical models.
- Nonhydrostatical models.

Hydrostatic models are models with hydrostatic approximations based on Eq. (1). These models contain parameterization of convection eliminating a major shortcoming in predicting of convective clouds (only intense showers and weaker storms that cannot cause a flash flood are modelled). Evaluated NWP models are ALADIN model for the Czech Republic and Slovakia, EURO4 and HIRLAM (Table 1).

Nonhydrostatical models are the most local with lower resolution and detailed topography of relief. These models are specialized to forecast of convective precipitation clouds. The main representatives are models GEM, WRF ARW and WRF NMM (Table 2).

Table 1. Parameters of hydrostatical NWP models [16, 17].

Models	ALADIN CR	ALADIN SR	EURO4	HIRLAM
Country of origin	Czech Republic	Slovakia Republic	GB	DE, EST, FIN, ICE, IR, HOL, NOR, SP, SWE, LIT
Resolution (km)	5 km	5 km	11 km	10 km
Area prediction	Czech Republic	Slovakia Republic	Europe	Europe
Time step	03, 06, 12, 24 h	03, 06, 12, 24 h	00, 05, 11, 17 h	00, 06, 12, 18 h
Time advance	2,5 days	3 days	2 days	3 days

Table 2. Parameters of nonhydrostatical NWP models [16, 17].

Models	GEM	WRF ARW	WRF NMM
Country of origin	France, USA, Canada	USA	USA
Resolution (km)	11 km	4 km	3 km
Area prediction	Europe	Europe	Europe
Time step	00, 12 h	00, 12 h	00, 12 h
Time advance	10 days	3 days	1 day

2.2 Algorithm of Convective Storm Prediction

The goal of the Algorithm is to provide an advanced information and prediction of heavy convective rainfall and dangerous phenomena, which may cause flash floods.

This algorithm is based on the analysis and evaluation of the prognostic meteorological variables and parameters from numerical weather prediction models.

The main output will be a report containing an assessment of the future development of convective precipitation systems in the 3 to 24 h in advance. Other prediction outputs are:

- Place of occurrence of convective precipitation for 13 municipalities with extended powers Zlin region and its 35 regions of municipalities with extended powers.
- Time of occurrence of convective precipitation with three-hour interval and
- The time predictions for 3–12 h, or 3–24 h in advance [17].

The probability and intensity of precipitation is calculated by the following equation:

$$P = \left(\sum n/\sum m \times 3\right) \times 100(\%), \tag{2}$$

where n is the sum of the sectional predictions and m is the total number of predicted parameters.

The Algorithm of convective storm predictions are composed of these steps (Fig. 1):

The partial forecasting steps are:

1. General characteristic contains basic information about the predicted situation (date, movement direction of precipitation, warning information and synoptic forecast).
2. NWP models - seven NWP model provides a forecast of precipitation for a given time interval.

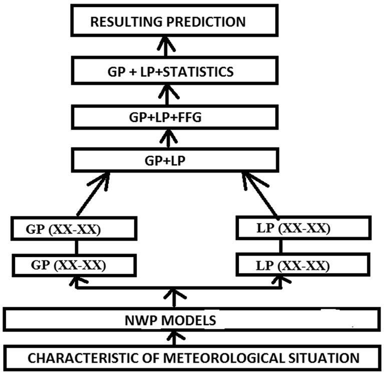


Fig. 1. The scheme of convective storm prediction

3. GP (xx-xx) is a global forecast for the three-hour interval predicted based on the second step. The data sources are ALADIN NWP models, GFS, WRF NMM and WRF ARW. The global forecast provides information on:
   - conditions of air mass above the condensation level and
   - storm (rainfall) intensity.
4. LP (xx-xx) is local forecast which gives information about conditions (potential triggers convection) of the earth's surface as the ground temperature, humidity, wind speed, cloud cover; pressure MSLP, orographic characteristics of relief). The data sources are ALADIN meteograms.
5. Statistics of historic storm situation includes a database of 100–200 weather situations. The resulting prediction is an intersection of Algorithm and selected statistic [18].

The main outputs of Algorithms of convective storm prediction:

- The probability (low/medium/high/very high) occurrence of convective precipitation computed from global and local predictions.
- The probability of occurrence of convective precipitation statistics of historical situations.
- Rainfall intensity (forecast of strong thunderstorms that may cause flash floods).
- The risk of flash floods determined by intersection of global, local predictions and soil saturation.
- The probability of occurrence of dangerous phenomena (heavy rainfall, hail, strong gusts and tornadoes).
- Type of convective storms of Global Forecasts (frontal/orographic/MCS/supercell convective storms).
- The probability of time and place the occurrence of precipitation by NWP models [18].

The probability of place of occurrence of precipitation, the risk of flash floods and more predictive outputs are classified on the calculated coefficients of convective precipitation forecast. Classification intensity of storms corresponds to the classification of dangerous phenomenon "Storm" by System of Integrated Warning Services of the Czech Hydrometeorological Institute.

2.3 Storms Prediction with the Use of Historical Data of Weather Situations

Storm prediction based on historical data meteorological situations is realized on the principle of estimation future state based on the previous state with the use a database of historical meteorological situations [19, 20].

Firstly, storm prediction is calculated from input data of NWP models. Subsequently, results are realized by artificial intelligence methods. The process of forecast creation is specified according to the below shown steps:

Table 3. Coefficients of rainfall intensity and probability occurrence of thunderstorms [17].

Coefficients	0	1	2	3
Intensity level	Weak thunderstorms	Strong thunderstorms	Very strong thunderstorms	Extremely strong thunderstorms
Rainfall intensity (mm/hours)	0–29	30–49	50–89	above 90
Probability of occurence (%)	0–24	25–49	50–74	75–100
Risk of flash flood	Low	Medium	High	Extremely high

1. Data collection.
2. Data processing and cultivation.
3. Data analysis.
4. Learning from data.
5. Development and testing of predictive models [19].

Data collection is performed from internal sources (data of 100–200 historical situations stored in MS Excel) and external sources (predictive data from NWP models) to a database of historical situations.

The specific service image of processing is the main tool for processing and cultivation data. The principle of this service is the conversion of image format into a set of parameters describing the meteorological or hydrological factors. The colour expression of the physical quantity of the factor is the converted parameter according to the scale factors (Table 3) [19].

Data analysis is addressed through statistical methods (Pearson correlation coefficient for comparison and finding dependencies between forecasted and historical weather elements and convection indices:

$$\rho_{X,Y} = \frac{E(XY) - E(X)E(Y)}{\sqrt{E(X^2) - E^2(X)}\sqrt{E(Y^2) - E^2(Y)}} \tag{3}$$

where stochastic quantities are $\mathrm{X} = \mathrm{E}\left(\mathrm{X}^2\right)$ and $\mathrm{Y} = \mathrm{E}\left(\mathrm{Y}^2\right)$. The correlation coefficient takes values in the range –1 to 1, where values approaching –1 are the least dependent and values approaching 1 increase addiction. Interval from 0.5 to 1 was experimentally chosen for the purpose of comparison [24].

Learning from data is a step where the used method of machine learning of neural networks. Used algorithm is the Backpropagation algorithm.

Input data are:

1. Meteorological elements specified conditions of air mass:
   – Temperature at altitude levels 1000, 925, 850, 700 a 500 hPa.
   – Relative humidity at altitude levels 1000, 925, 850, 700, 500 a 300 hPa.

- Wind direction at altitude levels 1000, 925, 850, 700, 500 a 300 hPa.
- Wind speed at altitude levels 1000, 925, 850, 700, 500 a 300 hPa.

2. Convection indeces (MLCAPE, MUCAPE, Lifted index, Showalter index, K-Index) (Fig. 2).

The resulting input value is determined by calculating the weighted average of these parameters. The transfer function is the logistic sigmoid. The outputs of predicted values are converted into coefficients according to Table 3.

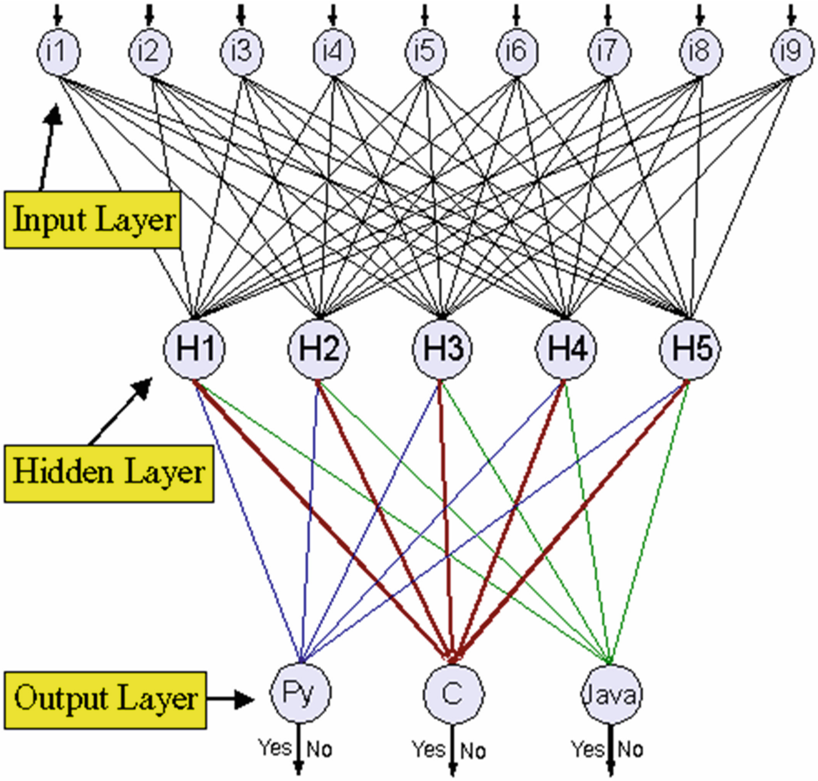


Fig. 2. Backpropagation algorithm [22].

3 Verification of Storms Prediction

Plenty of verification methods of weather forecasting are available. In practice, is often used the Skill Scores method based on the evaluation of the criteria in the pivot table. This method provides information on the number and frequency of cases where the phenomenon was or was not predicted and occurred or did not occur in all possible combinations [23]. The pivot table is adapted for calculating the percentages of success of convective precipitation forecasts:

Table 4. Contingency table for determination percentage values of success rate of forecast.

Criterion	Forecast	Reality	Result
HIT	1	1	1
MISS	0	0	1
FALSE ALARM	1	0	0
CORRECT REJECTION	0	1	0

Table 4 shows that the initial criteria represent a positive evaluation of predictions (coefficient 1, when the phenomenon occurred and was predicted). On the contrary, the last two criteria indicate bad prediction of the predicted or actual occurrence of the phenomenon (coefficient of 0, the phenomenon did not occur and was not predicted) [23]. Subsequently, the resulting percentage of the success rate of convective precipitation predictions is calculated according to the formula:

$$\bar{X} = \frac{1}{n}\sum\nolimits_{i=1}^{n} x_i \tag{4}$$

where $\sum$xi is the sum of the coefficients (0 or 1) of criteria for evaluating the success rate of convective precipitation forecasts, expressed as a percentage.

4 Discussion of Results

Percentage values of success rate of predictions were calculated for the determination of the probability of occurrence of convective precipitation:

- Algorithm of storm prediction.
- Algorithm Backpropagation.
- NWP models.

4.1 Evaluating of the Success Rate of Convective Precipitation Predictions for 13 Municipalities with Extended Powers Zlin Region

Firstly, the evaluation of the success rate of convective precipitation predictions was performed for the territory of 13 municipalities with extended powers in the Zlin Region. Success rates have been determined for the Algorithm of storms predictions, Backpropagation algorithm and NWP models. Assessed situations are historical situations in which the intensity of convective precipitation exceeded 20 mm/hr. with the probable occurrence of flash floods. The last two situations of 31.7 and 5.8 of the floods have caused considerable material damage in southwestern and southeastern part of Zlín Region.

As can be seen in Fig. 3 the lowest values predictions success rate was achieved in the first storm situation where both algorithms were deployed and tested for the first time. The success rate of forecast had a progressively upward trend where the maximum possible successful predictions has been reached in the past recent two flood

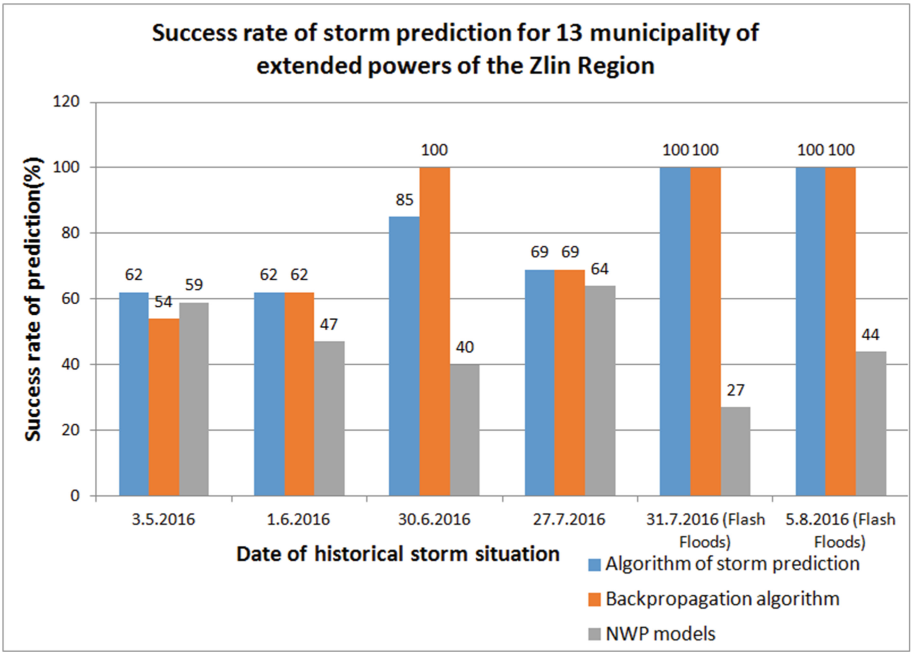


Fig. 3. Success rate of forecasting of convective precipitation for municipality with extended powers (MEP) in the Zlin region.

situation (such a high success rate was determined by the appearance of convective precipitation in the whole the Zlín Region). In contrast to both algorithms, which reached 80% of success rate, NWP models do not exceed the limit of 50% success rate predictions.

4.2 Evaluating of the Success Rate of Convective Precipitation Predictions for 35 Regions of Municipalities with Extended Powers Zlin Region

This evaluation includes the percentages success rate for only the Algorithm of storm prediction and Backpropagation algorithm. Prediction was experimentally chosen to MEP of 35 regions, wherein each MEP was divided into three regions. NWP models were not evaluated for the advanced prediction.

Figure 4 shows that the percentage of both prediction tools had an upward trend as in the first case. The success rate of forecasts reached an average from 60 to 70%, which is a relatively high value of success rate for a regionalized prediction.

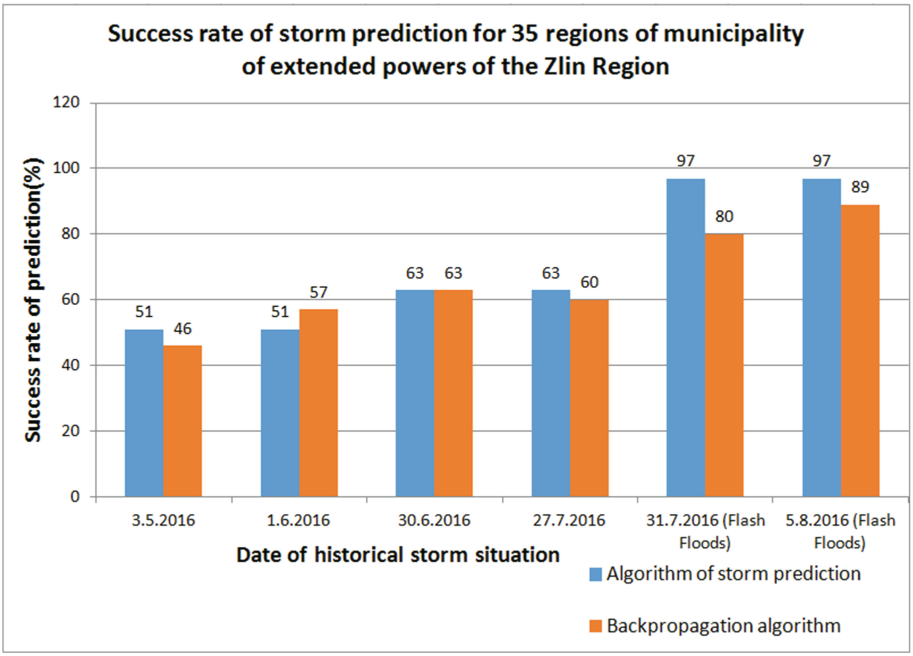


Fig. 4. Success rate of forecasting of convective precipitation for 35 regions of municipality with extended powers (MEP) in the Zlin Region.

5 Conclusion

The aim of the article was to evaluate the success forecasts storms through the NWP models, the Algorithm for the prediction of storms and the Backpropagation algorithm using an artificial intelligence. The success rate of forecasts was calculated for 35 regions including 13 municipalities with extended powers. The average success of both algorithms was around 60%, but NWP models amounted to only 47%. Consequently, the NWP models do not exceed the limit of 50%, so these predictive tools cannot be used for a qualified estimate of the probability of convective precipitation and flash floods. The algorithm storms prediction reached an average success rate of 75%, Backpropagation algorithm of 73%. Since the Backpropagation algorithm is part of the penultimate step of the prediction Algorithm so both algorithms can be used to prediction convective precipitation and the risk of flash floods.

Future research will focus on revising and optimizing forecasting parameters and their weights, including testing other algorithms of neural networks in order to achieve maximum success rate predictions about around 80% (the higher figure will not probably be achieved in terms of the quality of available data and shortcomings of NWP models). The main intention is to offer the Algorithm storms prediction at the Czech Hydrometeorological Institute for the inclusion and expansion of forecasting warnings on dangerous phenomenon “Storm” in the context of the Information Services of Warning System.

Acknowledgments. This work was supported by the Internal Grant Agency of Tomas Bata University under the project No. IGA/FAI/2017/019 “Information Support of Crisis Management at the Regional Level”.

References

1. Březková, L., Šálek, M., Novák, P., Kyznarová, H., Jonov, M.: New methods of flash flood forecasting in the Czech Republic. In: IFIP Advances in Information and Communication Technology, AICT, vol. 359, pp. 550–557 (2011). doi:10.1007/978-3-642-22285-6_59
2. Zdenek, S., Dusan, V., Jan, S., Ivan, M., Miroslav, M.: Protection from flash floods. In: Proceedings of the 26th International Business Information Management Association Conference - Innovation Management and Sustainable Economic Competitive Advantage: From Regional Development to Global Growth, IBIMA 2015, pp. 1359–1363 (2015). https://www.scopus.com/inward/record.uri?eid=2-s2.0-84976391745&partnerID=40&md5=923aa2f309578593d8b5e2cc503d02de
3. Hardy, J., Gourley, J.J., Kirstetter, P.-E., Hong, Y., Kong, F., Flamig, Z.L.: A method for probabilistic flash flood forecasting. J. Hydrol. **541**, 480–494 (2016). doi:10.1016/j.jhydrol.2016.04.007
4. Šaur D.: The Methodology Uses of Meteorological Radar of the Zlin Region for Crisis Management. Zlin, Czech Republic (2016)
5. Ravazzani, G., Amengual, A., Ceppi, A., Homar, V., Romero, R., Lombardi, G., Mancini, M.: Potentialities of ensemble strategies for flood forecasting over the Milano Urban Area. J. Hydrol. **539**, 237–253 (2016). doi:10.1016/j.jhydrol.2016.05.023
6. Jolivet, S., Chane-Ming, F.: WRF modelling of turbulence triggering convective thunderstorms over Singapore. In: Deville, M., Estivalezes, J.L., Gleize, V., Lê, T.H., Terracol, M., Vincent, S. (eds.) Turbulence and Interactions. Notes on Numerical Fluid Mechanics and Multidisciplinary Design, vol. 125, pp. 115–122. Springer, Heidelberg (2014). doi:10.1007/978-3-662-43489-5_14
7. Novák, P.: The Czech Hydrometeorological Institute’s Severe Storm Nowcasting System. doi:10.1016/j.atmosres.2005.09.014
8. Liechti, K., Panziera, L., Germann, U., Zappa, M.: The potential of radar-based ensemble forecasts for flash flood early warning in the Southern Swiss Alps. Hydrol. Earth Syst. Sci. **17**(10), 3853–3869 (2013). doi:10.5194/hess-17-3853-2013
9. Lakshmanan, V., Crockett, J., Sperow, K., Ba, M., Xin, L.: Tuning AutoNowcaster automatically. Weather and Forecast. **27**(6), 1568–1579 (2012). doi:10.1175/WAF-D-11-00141.1
10. Haiden, T., Steinheimer, M.: Improved Nowcasting of precipitation based on convective analysis fields. In: Precipitation: Advances in Measurement, Estimation and Prediction, pp. 389–417 (2008). doi:10.1007/978-3-540-77655-0_15
11. Beheshti, Z., Firouzi, M., Shamsuddin, S.M., Zibarzani, M., Yusop, Z.: A new rainfall forecasting model using the CAPSO algorithm and an artificial neural network. Neural Comput. Appl. **27**(8), 2551–2565 (2016). doi:10.1007/s00521-015-2024-7
12. Young, C.-C., Liu, W.-C., Chung, C.-E.: Genetic algorithm and fuzzy neural networks combined with the hydrological modeling system for forecasting watershed runoff discharge. Neural Comput. Appl. **26**(7), 1631–1643 (2015). doi:10.1007/s00521-015-1832-0
13. Chai, S.S., Wong, W.K., Goh, K.L.: Backpropagation vs. radial basis function neural model: rainfall intensity classification for flood prediction using meteorology data. J. Comput. Sci. **12**(4), 191–200 (2016). doi:10.3844/jcssp.2016.191.200

14. Meteorological Explanatory and Terminology Dictionary (EMS). Czech Meteorological Society (CMES), Prague. http://slovnik.cmes.cz
15. Batka, M.: Projections for the Development Atmosphere by Objective Methods. Prague, Czech Republic. http://kfa.mff.cuni.cz/wp-content/uploads/2015/03/kniha.pdf
16. WeatherOnline. http://www.weatheronline.cz/cgi-bin/expertcharts?LANG=cz&CONT=czcz&MODELL=gfs&VAR=prec
17. Šaur, D.: Comparison of success rate of numerical weather prediction models with forecasting system of convective precipitation. In: Silhavy, R., Senkerik, R., Oplatkova, Z. K., Silhavy, P., Prokopova, Z. (eds.) Artificial Intelligence Perspectives in Intelligent Systems. AISC, vol. 464, pp. 307–319. Springer, Cham (2016). doi:10.1007/978-3-319-33625-1_28
18. Šaur, D., Ďuricová, L.: Comprehensive system of intense convective precipitation forecasts for regional crisis management. In: The Tenth International Conference on Emerging Security Information, System and Technologies, SECURWARE 2016, IARIA, 24–28 July 2016, pp. 111-116 (2016). ISBN 978-1-64208-493-0
19. Predictive Analysis. https://www.gaussalgo.cz/prediktivni-analytika
20. Biological Algorithms (5) – Neural Networks: Learning – Backpropagation. https://www.root.cz/clanky/biologicke-algoritmy-5-neuronove-site/
21. Predictive Analysis. https://www.gaussalgo.cz/prediktivni-analytika
22. An Introducton to Neural Networks: Back-propagation. https://www.ibm.com/developerworks/library/l-neural/
23. Zacharov P.: Diagnostic and Prognostic Precursors of Convection. Faculty of Mathematics and Physics UK, KMOP, p. 61, Prague (2004). https://is.cuni.cz/webapps/zzp/detail/44489/
24. Calculation of the Pearson Correlation Coefficient. http://portal.matematickabiologie.cz/index.php?pg=aplikovana-analyza-klinickych-a-biologickych-dat–biostatistika-pro-matematickou-biologii–zaklady-korelacni-analyzy–pearsonuv-korelacni-koeficient–vypocet-pearsonova-korelacniho-koeficientu

Novel Decision-Based Modeling of Minimizing Usage Cost of Electricity in Smart Grid

A. Abdul Khadar[1(✉)], Javed Ahamed Khan[2], and M.S. Nagaraj[3]

[1] Department of E&E Engineering, BITM Ballari, Ballari, Karnataka, India
abkhadar416@gmail.com
[2] Department of E&E Engineering, MIT Madanpalli, Madanapalle, AP, India
[3] Department of E&E Engineering, BIET, Davangere, Karnataka, India

Abstract. The rapid growth of electricity consumption globally defines the need of electronic power conditioning, control of production and distribution of electricity over smart grid from a technical aspect. To reduce the global carbon emission, integrated framework deployments on the smart grid have gain lot of attention from research scientists. Therefore, the present scenario in the field of electricity and power distribution management incorporates the advanced information and communication technology. It can enhance efficiency, reliability and safety standards by conceptualizing distributed renewable energy utilization. This paper aims to represent an efficient and structured power distribution network (i.e. Grid) based framework to optimize the electricity cost of smart appliances. The proposed study conceptualizes the optimal framework by introducing a novel energy buffering methodology which integrates both micro grid controller and distributed energy storage to formulate the network. An analytical modeling has been introduced and tested considering storage capacity, financial cost, and electricity price to ensure the effectiveness of the proposed operational energy measurements using smart meters. The study also depicts ability of proposed method to significantly reduce the long-term term financial cost.

Keywords: Distributed energy storage · Optimal decision process · Micro grid controller · Smart grid

1 Introduction

The enormous power consumption due to the rapid growth of electricity usage in the major cities leads to a critical situation of the electricity crisis. To reduce the amount of carbon emission globally, the current research trends mostly focused into the smart grid technology deployment for efficient electricity and power distribution management. Therefore, research on smart grid has drawn wide attention from researchers. The state of art power distribution frameworks in smart grid is integrated with the information network, in order to efficiently distribute the renewable energy in terms of efficiency, reliability, and safety [1]. The smart grid technology is conceptualized in a way where imposing smart appliance enables self-configurable systems (i.e. commercial and residential electrical appliances) to regulate their power usage patterns automatically.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_14

The principle mainly follows the optimization of cost metric by reducing the power consumption during peak demand hours [2, 3]. It can be seen that micro-grid infrastructure can fully support the functionalities of smart appliances, which is basically located at the down-stream power distribution networked sub-station. The emerging trends on distributed power generations such as distributed energy resources e.g. windmills, turbines, solar panels and distributed energy storage has gained more and more concern in the recent years. The Fig. 1 shows a power distribution scheme with information network architecture consists of different type of components.

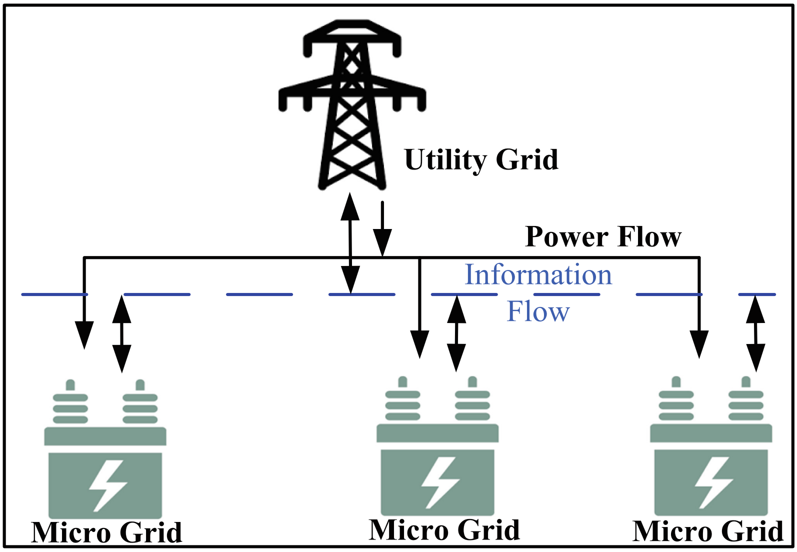


Fig. 1. Power distribution information network with micro grid

It was seen that most of the existing studies with adopted distributed energy storage are found exploiting the deliverable functionalities associated with irregular nature of power generation from the utility grid aspects. The conventional studies pertaining distributed energy storage haven't discussed that how to utilize it for minimizing the usage cost of smart appliances. Moreover, it is very difficult to make the best use of distributed energy storage resources considering optimized electricity price in real-time.

The proposed study discusses about an analytical modeling for minimizing the cost of electricity usage by the consumer using smart appliances running over grid architecture. The novelty of the proposed system can be seen by its feature of allocating the power to the respective smart appliances at the proper time with the lower financial cost [4, 5]. It also incorporates a Markov Optimal Process for incorporating a precise modeling of decision making. The comparative analysis shows the effectiveness of the proposed model. The organization of proposed paper is as follows Sect. 2 highlights the recent studies about the financial cost optimization, whereas Sect. 3 and four discusses the problem description and design methodology of the proposed model respectively, followed by result discussion (Sect. 5), Conclusion and future research direction is discussed in Sect. 6.

2 Related Work

This section discusses about the existing studies towards the power management techniques. The work carried out by Collotta and Pau [5] have presented a power management approximate for smart residences that integrates a wireless-network, stands on Bluetooth Low Energy, for communication among residence employments, with a residence power management plan. The recommended approximate addresses the collision of standby employments and high-power evaluation weights in peak hours to the energy expenditure charges of customers. Sun et al. [6] have presented a cognitive power technique for Communication-based train control schemes with smart-grids are illustrated to improve both train process presentation and cost effectiveness. The author also prepares a cognitive power scheme model for Communication-based train control methods. A review work carried out by Ye et al. [7] has presented a significance of a hierarchical up-link communication power control system for a neighborhood area network utilizing a two-stage "Stackelberg" amusement approximate. An illustrating on linear-receivers, the "Stackelberg" stability for the recommend game is obtained.

The research work carried out by Erol-Kantarci et al. [8] have discussed a complete study on the smart grid-driven approximates in an energy-efficient statements and data interiors and the communication between smart grid and data and announcement infrastructure. Ye et al. [9] have proposed cost-effective, elastic, and feasible neighborhood area network design utilizing wireless equipments such as *"IEEE 802.11 s" and "IEEE 802.16"*. They presented an optimization issue to reduce overall rate. To resolve the issue, they also studied the issue of choosing the best number of access in a neighborhood area network. The review work carried out by Martín-Arias [10] has discussed about a superior idea of lighting method intend for urban surroundings with high-customization abilities with the accumulation of lighting approach directed through wireless methods. Bera et al. [11] have demonstrated an energy competent smart indicating system towards reducing the power expenditure by the smart indicators for green smart network statement.

Wan et al. [12] have presented an unique technique for predicting the energy management system controlled by smart grids. The authors have designed the unique charecteristics of solar power as well as photovoltaic power followed by standardizing the performance values. The design is completely done using statistical mechanism. This work has also elaborated about other form of forecasting models in power management with a special focus on its merits and demits. Pang et al. [13] have illustrated a effective smart metering approximate as an essential piece of building automation and control system for the insist manage of construction examines. It may help customers to observe the energy expenditure and take necessary action to reduce their energy competence at a better granularity. The review study transmitted by Ortiz et al. [14] have presented an easy but yet capable network innovation and tree building protocol that also acts as an smart metric for direction selection in smart surroundings. The author has also emphasized about usage of fuzzy-logic. They also carried out experimental analysis to prove that presented system offer better efficiency in contrast to conventional routing parameters.

Hence, it can be seen that there are quite a good amount of work being carried out towards smart grid system. The next segment discusses about the issues.

3 Problem Description

In the recent times, enormous usage of residential and commercial smart appliances leads to indefinite power consumption. Therefore a situation arises where dealing with financial cost of electricity usage has become more challenging. Most of the state of art power distribution management system does not discusses about the financial cost effectiveness from the technical perspective whereas very fewer studies are found to have significant impacts on reduction of peak demands and lowering risk of grid operations. One of the most significant research problem is that there are lesser amount of work towards exploring effectiveness of the power evaluation system in recent times. It has not been well investigated for the micro-grid that how to utilize the distributed energy storage to decrease the financial price of smart appliances users. It is very challenging to make best use of the valuable distributed energy storage resource with the consideration of real-time electricity price, as well as the characteristics of the distributed energy storage and smart appliances.

4 Design Methodology

This section of the paper introduces the design methodology of the proposed framework namely Optimal Decision Process in smart grids to effectively reduce the electricity cost of smart appliances. The proposed model is conceptualized based on a theoretical modelling of an optimal scheme which incorporates Markov Decision Process for token allocation policy under distributed energy storage scenario.

The following Fig. 2 shows the architecture of the proposed framework. It initiates a set of operations which include Micro-Grid Controller (MCG) operations, Smart Appliances (SA) operations, and Distributed Energy Storage (DES) power pricing and allocation policy. The proposed framework considers a micro-grid controller module which is further integrated with SAs and a distributed information network, which could be possibly a neighborhood area network or building area network. However, the proposed system introduces the frame structure of a micro-grid controller networked model which is primarily composed of a set of decision and action phases. The highlighted Fig. 2 shows that proposed framework imposed different type of operations to reduce the financial costs associated with each smart appliance. It incorporates a principal of bursty energy usage patterns which could possibly minimize the overall monetary cost of smart appliance. The optimal decision problem has been derived using Markov Decision process. The following Fig. 3 highlights the integrated

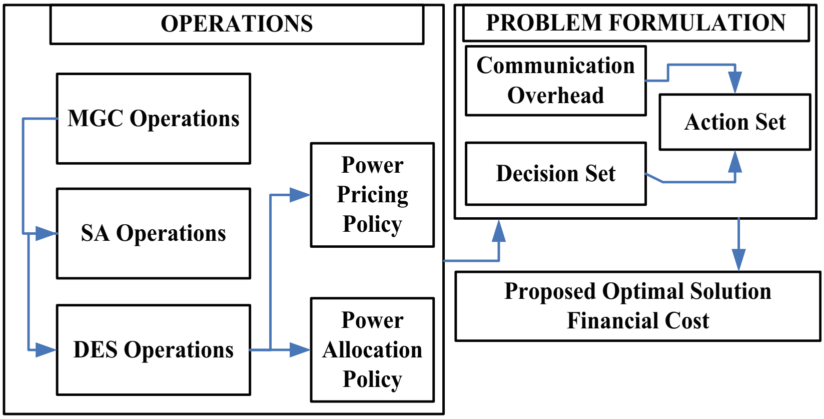


Fig. 2. Proposed system architecture

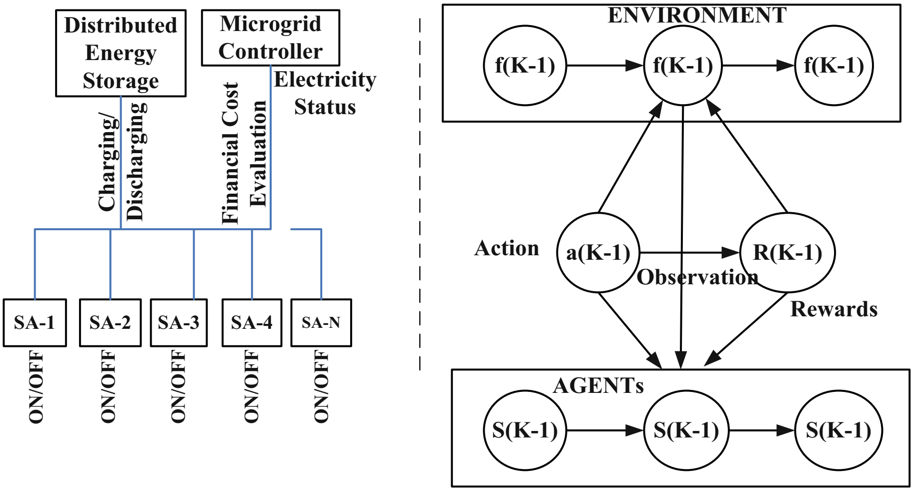


Fig. 3. Integrated Markov Decision Process

framework where Markov Decision process provides an efficient decision making to evaluate distributed energy storage power pricing and allocation policy.

The analytical modeling for distributed energy storage information set, Decision and Action set is presented below.

(i) **Information Set**

The information is defined by the following mathematical expression

$$\beta(k) = \{p(k),\ d(k),\ m(k),\ l(k)\} \tag{1}$$

where p(k) denotes the current electricity price per energy unit the vector d(k) represents the energy demand states it is also represented by $d(k) = [d_1(k), d_2(k)\ldots. d_M(k)]^T \in B^M$ where B = {0,1} and M is the index associated with Micro-grids and SAs.

(ii) **Decision and Action Set**

The distributed energy storage decision set is derived by the following mathematical expression

$$\eta(k) = \{a(k),\ t(k)\} \tag{2}$$

And the action set is defined in the following Eq. (3)

$$a(k) = \left\{ \begin{matrix} 1 \\ 0 \\ -i \end{matrix} \right\} \tag{3}$$

1 will be considered when distributed energy storage will be in charging state similarly, 0 denotes the idle state and –i considers discharging scenario where μ_i where i = {1, 2, 3}. An optimization process also carried out to reduce the financial cost which is basically derived by the following equation.

$$\lim_{L \to \infty} \text{Min}\ (\eta\ (\ k),\eta(k)) \tag{4}$$

The next section discusses about the results obtained by extensive simulation carried out in MATLAB environment.

5 Results and Discussion

The proposed framework is executed in Matlab considering an objective to decrease the financial price of electrical appliances in smart grid. The simulation framework considers the input parameters such as delay constraints, distributed energy storage capacity which have been considered in between (5–20) and high price state in USD. It also considers the number of frames 50. The results obtained from the simulation environment shows that the proposed optimal framework can significantly reduce the financial cost of electrical appliances in smart grid.

The above Fig. 4 shows the financial cost obtained by simulating three different baselines such as BL-1, BL-2, and Myopic considering delay constraints which are further compared with the values obtained by simulating the proposed optimal framework. Therefore the comparative analysis depicts that the proposed model achieves optimal and minimized financial cost in comparison with baselines.

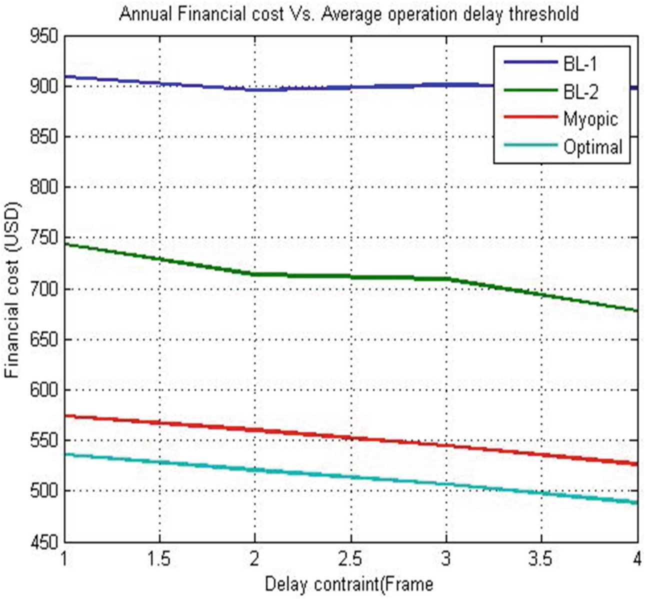


Fig. 4. Annual Financial cost Vs Average operational threshold

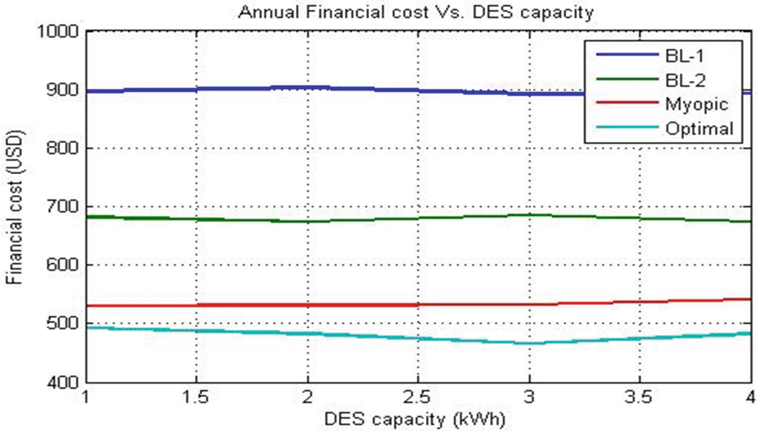


Fig. 5. Annual Financial cost Vs DES Capacity

The Fig. 5 shows that the optimal framework reduces the long term financial cost (USD) significantly with respect to distributed energy storage capacity (kWh). The performance improvement of the proposed optimal framework has been carried out by incorporating distributed energy storage power allocation and pricing policy which imposed an optimal Markov decision making process for efficient power allocation and distribution over an information network. The comparative analysis depicts the effectiveness of the proposed system.

6 Conclusion and Future Research Work

In this paper, an optimal framework for reducing long term financial cost of electrical appliances was proposed for deferrable smart appliances. The proposed modeling determines distributed energy storage charging/discharging actions and the distributed energy storage power allocation policy with the consideration of their expected impact on the future cost. This paper discussed a novel approach of modeling energy buffering. Simulation results showed that the proposed optimal framework outperforms the conventional baselines by reducing the long term financial cost efficiently. It also shows the effectiveness of the proposed optimal framework by taking various performance parameters such as financial cost, distributed energy storage capacity and delay constraint into consideration.

References

1. Farhangi, H.: The path of the smart grid. IEEE Power Energy Mag. **8**(3), 18–28 (2010)
2. Huang, A.Q., Crow, M.L., Heydt, G.T., Zheng, J., Dale, S.: The Future Renewable Electric Energy Delivery and Management (FREEDM) system: the energy internet. Proc. IEEE **99** (1), 133–148 (2011)
3. He, M., Murugesan, S., Zhang, J.: Multiple timescale dispatch and scheduling for stochastic reliability in smart grids with wind generation integration. In: Proceeding of IEEE INFOCOM, pp. 461–465 (2011)
4. Roberts, B.P., Sandberg, C.: The role of energy storage in development of smart grids. Proc. IEEE **99**(6), 1139–1144 (2011)
5. Collotta, M., Pau, G.: A novel energy management approach for smart homes using bluetooth low energy. IEEE J. Sel. Areas Commun. **33**(12), 2988–2996 (2015)
6. Sun, W., Yu, F.R., Tang, T., You, S.: A cognitive control method for cost-efficient CBTC systems with smart grids. IEEE Trans. Intell. Transp. Syst. **PP**(99), 1–15
7. Ye, F., Qian, Y., Hu, R.Q., Das, S.K.: Reliable energy-efficient uplink transmission for neighborhood area networks in smart grid. IEEE Trans. Smart Grid **6**(5), 2179–2188 (2015)
8. Erol-Kantarci, M., Mouftah, H.T.: Energy-efficient information and communication infrastructures in the smart grid: a survey on interactions and open issues. IEEE Commun. Surv. Tutorials **17**(1), 179–197 (2015)
9. Ye, F., Qian, Y., Hu, R.Q.: Energy efficient self-sustaining wireless neighborhood area network design for smart grid. IEEE Trans. Smart Grid **6**(1), 220–229 (2015)
10. Martín-Arias, M., Huerta-Medina, N., Rico-Secades, M.: Using wireless technologies in Lighting Smart Grids. In: International Conference on New Concepts in Smart Cities: Fostering Public and Private Alliances (SmartMILE), Gijon, 2013, pp. 1–6 (2013)
11. Bera, S., Misra, S., Obaidat, M.S.: Energy-efficient smart metering for green smart grid communication. In: IEEE Global Communications Conference, Austin, TX, pp. 2466–2471 (2014)

12. Wan, C., Zhao, J., Song, Y., Xu, Z., Lin, J., Hu, Z.: Photovoltaic and solar power forecasting for smart grid energy management. CSEE J. Power Energy Syst. **1**(4), 38–46 (2015)
13. Pang, C., Vyatkin, V., Deng, Y., Sorouri, M.: Virtual smart metering in automation and simulation of energy-efficient lighting system. In: IEEE 18th Conference on Emerging Technologies & Factory Automation (ETFA), Cagliari, pp. 1–8 (2013)
14. Ortiz, A.M., Royo, F., Olivares, T., Timmons, N., Morrison, J., Orozco-Barbosa, L.: Intelligent routing strategies in wireless sensor networks for smart cities applications. In: 10th IEEE International Conference on Networking, Sensing and Control (ICNSC), Evry, pp. 740–745 (2013)

Integrated Algorithm of the Domain Ontology Development

Viktor Kureichik and Irina Safronenkova(✉)

Autonomous Federal State Institution of Higher Education,
Southern Federal University, Taganrog, Russia
kur@tgn.sfedu.ru, safronenkova050788@yandex.ru

Abstract. Due to global computerization of society and heavy increase of information flow ontologies found a use in many fields concerning with big data processing. A great number of computer applications use ontology as a tool of knowledge representation. However, the problem of ontology development is still difficult and time consuming task. This paper demonstrates a new integrated algorithm of CAD tasks ontology development. Two experiments of domain ontology development were performed. The result is visualized in Protégé 4.2. The developed ontology intends the further modifications by the researchers from all over the world. This allows improve efficiency of knowledge processing and classification of large data arrays of a specific domain.

Keywords: Ontology · Attributes · Class · Instances · Concepts · CAD-system · Bayesian classifier · Hierarchical clustering

1 Introduction

Problems concerned with the processes and methods of searching, capturing, storage and processing of information are topic of interest and demand new solution approach. Information technology helps to solve these problems. The role of ontologies can hardly be overestimated in information and computer science. In recent decades ontology has become a synonym for the solution of many problems concerning with big data processing. At the present time research scientists and network users deal with tasks which solved with the help of ontology successfully. Such tasks include classification and clustering of documents, developing of vocabulary and thesaurus, Semantic Web, decision problem and knowledge acquisition [1]. As can be seen from the above ontology is a kea decision of problems of processing of large data arrays considering semantics. Here a crucial criterion is speed and low price of ontology development. But in spite of ubiquity of ontology usage, the process of ontology development is time consuming, expensive and demands work of experts group. For this reason automated ontology development is an actual task at the moment. The given paper gives a new integrated algorithm of automated CAD tasks ontology development. This algorithm allows creating new domain ontologies and modifying existing ones. Two experiments of domain ontology development demonstrate an opportunity of reducing the development time and extension of initial domain ontology by adding classes, subclasses, instances and concepts.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_15

2 Ontologies

Before discussing the problem of automated ontology development it's necessary to give a definition of the term "ontology". There are a number of definitions for this term. The Webster dictionary [2] defines the term ontology as:

(D1). a branch of metaphysics concerned with the nature and relations of being;
(D2). a particular theory about the nature of being or the kinds of existents;
(D3). a theory concerning the kinds of entities and specifically the kinds of abstract entities that are to be admitted to a language system.

In information technologies the term ontology denotes explicit specification of conceptualization, where conceptualization presents a description of variety of objects and relations between them [3]. An ontology must be formal and, therefore, machine readable. This way ontologies can provide a common vocabulary between various applications [4]. Formal ontology model is a cortege with length equals three:

$$O = \langle M, \psi, F \rangle \tag{1}$$

where equation F is a finite set of interpretation functions given on the bases of concepts and/or relations of ontology; M is a finite set of a domain concepts; ψ – is a finite set of relations between ontology concepts $\Psi = \{\varphi_1, \varphi_2, .., \varphi_n\}$ [5, 6].

Formal relation R between the sets S and T is a subset of the cartesian product $S \times T$. Besides binary relations one can also consider n-ary relations with $n \geq 0$. An n-ary relation R on the sets $S1, \ldots, Sn$ is a subset of the cartesian product $S1 \times \cdots \times Sn$. The identity relation "=" is a relation on a set S. This relation is often denoted by I. So, $I = \{(s, s) \,|\, s \in S\}$ [7]. As is known, relation is connectivity between objects in nature. We can notice a lot of common between concepts in ontology and relations in discrete mathematics. Given an element of domain ontology that is a set $< a, b > \ \in O$, where a, b – domain concepts. If a set $< a, b > \ \in O$ then a relation $a\, a\, \varphi\, b$ exists where concepts a and b are in a certain connection. In other words a concept a is in relation R to be b. If a, b – domain concepts $(a,\ b \in M)$, then $a\ \varphi\ b$ is a true or false statement. $\varphi = \ < O, M >$ is a complete relation if $O = M^2$, i.e. $(\forall a, b \in M)(a\, \varphi\, b)$. $\varphi = \ < O, M >$ is an empty relation if O is an empty set, i.e. $\varphi = \ < \emptyset,\ M >$.

Generally ontologies consist of classes, instances, concepts, attributes and relations. Instances are the basic components of low-level ontology. They may represent any physical and abstract objects. Concepts or classes are groups or sets of other objects. Instances of ontology often have attributes. Every attribute has a name, value and store information about the instance. Classes may have subclasses including of combinations of other classes, instances or concepts [8].

Ontologies can be classified into three classes according to their level of formalization. A formal ontology is a conceptualization whose categories are distinguished by axioms and definitions. Such type of ontologies are stated in logic that can support complex inferences and computations. The second type is prototype-based ontologies whose categories are distinguished by typical instances or prototypes rather than by axioms and definitions in logic. Categories are formed by collecting instances extensionally rather than describing the set of all possible instances in an intensional way, and selecting the most typical members for description. For their selection, a similarity metric on instance terms are applied. Terminological ontologies are partially specified by subtype-supertype relations and describe concepts by concept labels or synonyms rather than prototypical instances, but lack an axiomatic grounding. Terminological and prototype-based ontologies cannot be used in a straightforward way for inference, but are easier to construct and to maintain [9].

3 Ontology Development

One of the fundamental aspect of ontology development is a type of input data. It can be divided into three groups:

- structured data: database schemes.
- semi-structured data: dictionaries like WordNet;
- unstructured data: natural language text documents, like the majority of the HTML based webpages [10].

The process of ontology development can be divided in six stages (Fig. 1). To define the notions of domain (a set M) it's necessary to identify natural language terms that refer to them. Synonym identification helps to avoid concept redundancy which is widespread in unstructured data. Now a set of terms is defined and serves as a source of concepts. Next step is building concept hierarchies, in other words it's necessary to identify a taxonomic relationship between notions. Then it's also necessary to extract non taxonomic relations (a set Ψ). These relations show an additional semantic aspects in ontology. Finally, is the step of instances extracting which populate the domain ontology. Significantly to extract axioms for deriving facts that are not explicitly expressed in the ontology (a set F).

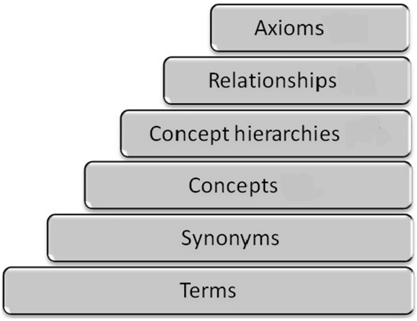


Fig. 1. Ontology development process [11]

4 Integrated Algorithm

Knowledge acquisition is the most crucial task for ontology developers. Although the unstructured data is the most widespread format of information representation nowadays (PDF, Word, Web pages), it is still the most difficult for machine reading.

Problem definition: to develop CAD tasks ontology automatically by knowledge acquisition from a set of natural language text documents.

This work represents a new integrated algorithm of ontology development. It's based on two approaches of machine learning. The algorithm is shown in Fig. 2.

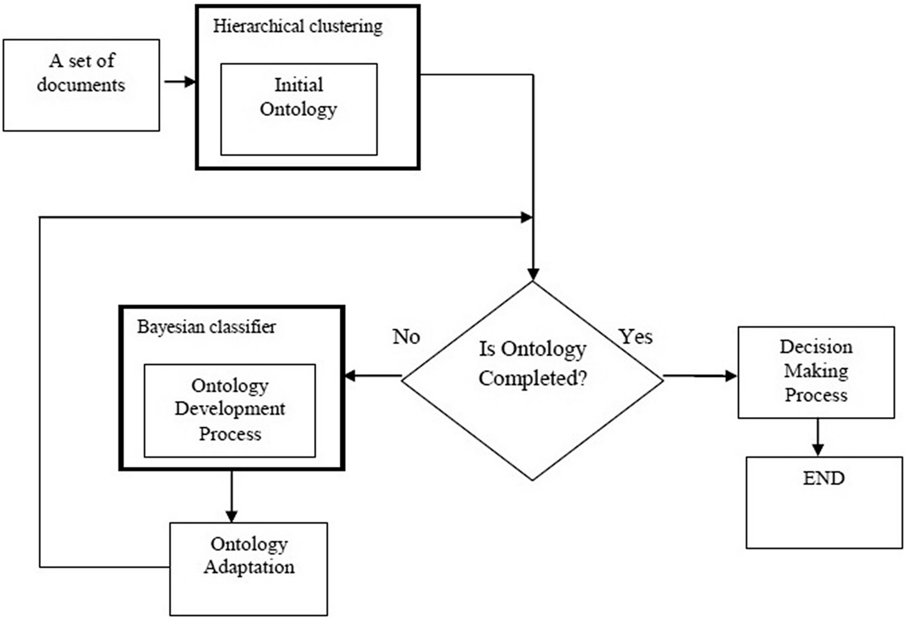


Fig. 2. Integrated algorithm of ontology development

The first step is development an initial domain ontology by hierarchical clustering from a set of natural language text documents. Clustering can be defined as the process of organizing objects into groups whose members are similar in some way based on the distributional representation [12]. The main idea of this step is that term sets can be organized into a hierarchy that can be transformed directly into an ontology prototype. In general there are three major styles of clustering:

1. Agglomerative: In the initialization phase, each term is defined to constitute a cluster of its own. In the growing phase larger clusters are iteratively generated by merging the most similar/least dissimilar ones until some stopping criterion is reached.
2. Partitional: In the initialization phase, the set of all terms is a cluster. In the refinement phase smaller clusters are (iteratively) generated by splitting the largest

cluster or the least homogeneous cluster into several subclusters. In practice, partitional clustering (like the K-Means clustering technique) is faster as it can be performed in runtime of O(n) compared to O(n2) for agglomerative techniques, where n is the number of represented terms.

3. Conceptual: Conceptual clustering builds a lattice of terms by investigating the exact overlap of representing terms between two represented terms. In the worst case, the complexity of the resulting concept lattice is exponential in *n* [13].

To demonstrate an agglomerative clustering technique let consider an example of CAD tasks ontology development.

Experiment 1

Let there be given a set of CAD tasks of four {Task1, Task2, Task3, Task4} and a set of properties of two {p1, p2} each of which can belong to any task. In this case a property defines a number of certain words which can be found in every task (Table 1).

Table 1. A table for clustering

CAD tasks	P1	P2
Task 1	1	2
Task 2	2	2
Task 3	3	4
Task 4	4	1

Let Task 1 = A, Task 2 = B, Task 3 = C, Task 4 = D. Let an abscissa axis is a property 1, an ordinate axis is a property 2. In this work a property is an attribute of a concept, the concept is a CAD task. These concepts form a cluster or a class. It is clearly shown in diagram (Fig. 3).

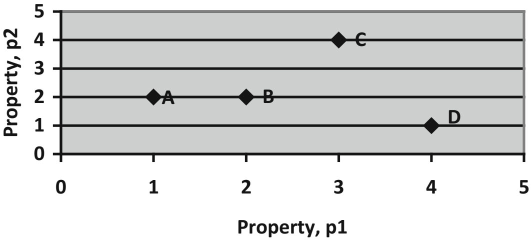


Fig. 3. Diagram of CAD tasks

At the first stage each element is a cluster. As is seen from diagram the closet elements are A and B which will join the first bigger cluster. In such way the clustering is continuing. The result of agglomerative clustering technique can be visualized with the help of Protégé 4.2 (Fig. 4).

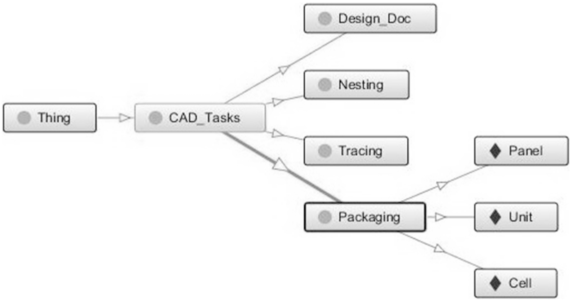


Fig. 4. CAD tasks ontology

In Fig. 4 we can see four classes that present design phases. They are “Packaging”, “Nesting”, “Tracing” and “Design Documents”. The class “Packaging” has instances “Cell”, “Panel” and “Cell” that show different types of packaging.

Then it’s a step of decision making process. An experts group has to decide if the initial ontology is completed or need to be modify. The question of the ontology evaluation is an actual and hard for ontology engineering nowadays. There are a lot of existing methods of ontology evaluation. Here the main ones:

- completeness and accuracy of the vocabulary;
- adequateness of the ontology taxonomy and relations;
- perceptibleness;
- productivity in applications;
- selection one ontology among existing ones [14].

It is important to notice that the choice of the method of ontology evaluation is individual for each case and depends on many factors.

If the domain ontology completed and customized the algorithm is finished. If not the algorithm has to modify the initial ontology by Bayesian classifier. Bayesian classifier is a simple probabilistic classifier based on applying Bayes theorem. Bayesian classifier is based on the idea that the role of a class is to predict the values of features for members of that class. Examples are grouped in classes because they have common values for the features. Bayes theorem is stated mathematically as the following equation:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{2}$$

where P(A) and P(B) are the probabilities of observing A and B without regard to each other;
P(A|B), a conditional probability, is the probability of observing event A given that B is true;
P(B|A) is the probability of observing event B given that A is true [15].

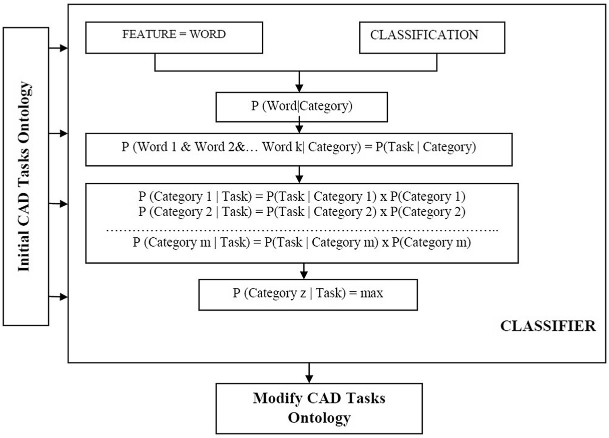

Initial CAD Tasks Ontology
FEATURE = WORD
CLASSIFICATION
P (Word|Category)
P (Word 1 & Word 2&… Word k| Category) = P(Task | Category)
P (Category 1 | Task) = P(Task | Category 1) x P(Category 1)
P (Category 2 | Task) = P(Task | Category 2) x P(Category 2)
P (Category m | Task) = P(Task | Category m) x P(Category m)
P (Category z | Task) = max
CLASSIFIER
Modify CAD Tasks Ontology

Fig. 5. A pattern of Bayesian classifier

The technique can classify almost any sort of data. In Bayesian classifier, the learning agent builds a probabilistic model of the features and uses that model to predict the classification of a new example.

This paper presents an ontological approach for Bayesian classifier training. The idea is based on ontology use instead of training set. Primary advantage of this approach is no need for developing a training set and further ontology use for different purposes. Let see an example of modifying CAD tasks ontology by Bayesian classifier. Experiment 2

Let there be given an initial CAD tasks ontology consists of *m* categories. It's necessary to modify it with the help of Bayesian classifier (Fig. 5). In our case an attribute is a number of the certain words in the CAD task, a concept of ontology is a category of the CAD tasks. According to the pattern, it's necessary to estimate a conditional probability that a word falls into any category of classification. The next step is calculating a probability that a task falls into any category. By the Bayes theorem we can find a conditional probability that a task falls into the category. Then Bayes theorem is stated mathematically as the following equation:

$$\mathrm{P(Category|Task)} = \frac{\mathrm{P(Task|Category)P(Category)}}{\mathrm{P(Task)}} \tag{3}$$

where P(Category | Task) is a probability that a concrete task falls into a concrete category;
P(Category | Task) was calculated earlier;
P(Category) is the quotient of tasks relating to a certain category by the total number of tasks;
P(Task) is the same value for all categories.

This probability calculates for each category and then a maximum of probability is found [16]. A pattern of Bayesian classifier is shown in Fig. 5.

In the case of ontological approach it's necessary to calculate a value R_{pk} - is a degree of a membership a task p to a category k. This calculated by q_{pk} – is a number of domain concepts appearance (k) in the task p, = 1, 2,…n; k = 1, 2,…m:

$$R_{pk} = \frac{q_{pk}}{\sum_{i=1}^{m} q_{pi}} \tag{4}$$

where $q_{pk} = \sum_{j=1}^{M_j} Q_{jpk}$;

Q_{jpk} – a number of domain concepts (j) appearance in the task p;
M_j –a number of concepts in domain ontology k.

When a new set of CAD tasks enters on a classifier input, it may be that some instances will not be classify by existing ontology. Then new subclasses and instances may be obtained. This process is also manual and executed by experts group.

The result of ontology adaptation is shown in Fig. 6. As is seen from the Fig. 6 a modify CAD tasks ontology has new subclasses and instances than the initial one. A class "Tracing" has subclasses "Wire_routing", "Order_connections", "A_list_of_connections" and "Layer_nesting". These subclasses name the stages of tracing. Subclass "Wire_routing" has instances: "Channel_routine", "Beam_routine", "Topological_routine" and "Wave_routine". They present solution algorithms of wire routing.

A phase of ontology adaptation can be repeated for as long as it will need for a good result. When the purpose which is defined by a customer is achieved the algorithm is finished.

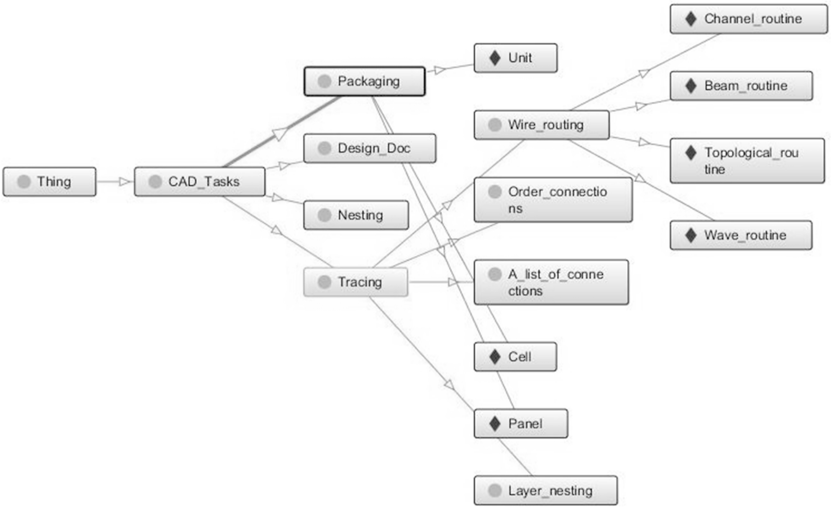


Fig. 6. Modify CAD tasks ontology

5 Conclusion

The paper presents a new integrated algorithm of domain ontology development. This algorithm combines supervised and unsupervised methods: hierarchical clustering and Bayesian classifier. An example of CAD tasks ontology is given and visualized in Protégé 4.2. The experiments show the main advantages of the integrated algorithm which are:

1. multiply remodel of initial ontology;
2. ontology reusability;
3. no need to develop a training set;
4. variations of developed ontologies depending on a set of input data;
5. quality input and speeding of ontology developing;
6. substantial reduction of charge.

Acknowledgments. We thank collaborators of automated engineering system department for work results discussion. This research is provided by the Russian Foundation for Basic Research through grant #2.5537.2017/ BУ in South Federal University

References

1. Kureychik, V.M., Semenova, A.V.: Domain ontology development for linguistic purposes. In: 9th International Conference on Application of Information and Communication Technologies, pp. 83–87. IEEE Press, Rostov-on-Don (2015)
2. Webster dictionary. http://www.webster-dictionary.net
3. Dobrov, B.V., Ivanov, V.V., Lukashevich, N.V., Solov'ev, V.D.: Ontology and Thesaurus: Models, Tools, Applications. BINOM, Moscow (2013)
4. Gruber, T.: Toward principles for the design of ontologies used for knowledge sharing. Int. J. Hum. Comput. Stud. **43**(4–5), 907–928 (1995)
5. Kagalovsky, M.R.: Conceptual and ontological modeling in information systems. Program. Comput. Softw. **35**(5), 241–256 (2009)
6. Kureychik, V.M., Safronenkova, I.B.: Creation of CAD-systems ontology using Protege 4.2. In: All-Russia Science&Technology Conference "Problems of Advanced Micro- and Nanoelectronic Systems Development", vol. 3, pp. 240–245. IPPM RAN (2016)
7. Cuypers, H.: Discrete Mathematics. Springer, Heidelberg (2007)
8. Noy, N., McGuinness, D.: Ontology development 101: a guide to creating your first ontology. Stanford Knowledge Systems Laboratory Technical report KSL-01-05 and Stanford Medical Informatics Technical report SMI-2001-0880 (2001)
9. Drumond, L., Girardi, R.: A survey of ontology learning procedures. In: Proceedings of the 3rd Workshop on Ontologies and their Applications. CEUR Workshop Proceedings, vol. 427 (2008)
10. Biemann, C.: Ontology learning from text: a survey of methods. LDV-Forum **20**, 75–93 (2005)
11. Introduction to Ontology Learning. http://www.jens-lehmann.org/files/2014/pol_introduction.pdf

12. A Tutorial on Clustering Algorithms. http://home.deib.polimi.it/matteucc/Clustering/tutorial_html/
13. Maedchen, A., Staab, S.: Ontology learning. In: Staab, S., Studer, R. (eds.) Handbook on Ontologies Euro-Par 2004. International Handbooks on Information Systems, vol. 2, pp. 173–190. Springer, Heidelberg (2004). doi:10.1007/978-3-540-24750-0_9
14. Gavrilova, T.A., Kudryavtsev, D.V., Muromtsev, D.I.: Knowledge Engineering: Models and Methods. St. Petersburg, Lan' (2016)
15. Bayesian Inference. https://cran.r-project.org/web/packages/LaplacesDemon/vignettes/BayesianInference.pdf
16. Kureychik, V.M., Safronenkova, I.B.: Automated classification cad-system tasks using domain ontology. In: Conference on Artificial Intellegence «CAI 2016», vol. 2, pp. 216–223. Universum, Smolensk (2016)

Formal Models of the Structural Errors in the Knowledge Bases of Intellectual Decision Making Systems

Olga Dolinina(✉) and Natalya Suchkova

Yury Gagarin State Technical University of Saratov, Saratov, Russia
odolinina09@gmail.com, rinoa_27@mail.ru

Abstract. The paper provides classification of structural errors in the rule-based knowledge bases of intellectual decision making systems. For detecting of the structural errors it is proposed to use AND/OR graph representing a knowledge base of rule-based system. There are described formal models of the 3 types of structural errors identified: redundancy errors, incompleteness errors and inconsistency. Redundancy errors are described with duplicates, redundant inference chains, insignificant inference chains, incorrect inference chains and cycles. Incompleteness error example is isolated vertices. Inconsistency in knowledge bases is represented by conflicting inference chains. For each structural error the paper provides formalization in terms of the graph model and the way of correction.

Keywords: Debugging of the intellectual decision making systems · Rule-based systems · Static analysis · Verification · Structural errors · AND/OR graph

1 Introduction

Intellectual decision making systems (IDMS) are used in wide range of areas: industry, medicine, research activities, education, classification tasks including critical areas. It demands reliability of making decisions by these systems what depends on all components of the IDMS. Methods and algorithms of improvement of the reliability of the software and hardware components are well developed but providing of the quality of the knowledge base (KB) is still a problem which has not been solved yet. Methods of knowledge bases (KB) debugging are still not formalized. Algorithms of the traditional software debugging cannot be used for the knowledge bases and most of the developers use expert approach for the debugging of the knowledge bases.

Quality of KB is a multi-criteria problem, there are different approaches for the checking of the correctness and completeness including methods of detecting of various types of errors [1–7], but debugging of the KB is still considered to be the most complicated stage of the IDMS development. There are errors in the knowledge base which are connected with the inconsistency of the knowledge area – for example, errors of forgetting-about-the-exception type [2] and which can be detected by the testing only. At the same time, so called, structural errors can be detected and deleted from the

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_16

KB on the stage of the static debugging (the formal checking of the KB) which does not demand the running of the intellectual system.

The absence of the structural errors does not guarantee the absence of the errors connected with the inconsistency of the knowledge area but it increases the effectiveness of the decision making due to the reducing of the solution time, so the first necessary step of the debugging of the KB is the formal checking (static analysis).

In the papers [3–7] several structural errors in the KB and algorithms for their detecting are described but no errors formalization has been made and the current paper accumulates full information on structural errors and provides formal models of the errors which can be used as a basis for automatic verification of knowledge bases structure.

2 Formal Models of Structural Errors of the Rule-Based Knowledge Base

Rules are widely used for representing of the knowledge bases of intellectual decision making systems. Rule-based knowledge base is set as:

$$P = (F, R, G, C, I), \tag{1}$$

where F is a finite set of the facts in the concrete field about the problem.

R – a set of rules where

$$r_m : IF\, f_i\, and\, f_j \ldots and\, f_n\, then\, f_k, \tag{2}$$

G – set of goals or the IDMS terminal facts; C is the set of the permitted combination of the facts; I – the interpreter of the rules, realizes the goal solution.

Let S is the set of input facts, i.e. facts specified by the user in the input of the intellectual system. $S \subset F$.

The logic of the knowledge base can be presented by the AND/OR graph. For example, let's build AND/OR graph for a set of the following rules:

$r_1 : if\, s_1\, and\, s_2,\, then\, f_1;$
$r_2 : if\, s_2\, and\, s_3,\, then\, f_2;$
$r_3 : if\, s_3\, and\, s_4\, and\, s_5,\, then\, f_3;$
$r_4 : if\, f_1,\, then\, g_1;$
$r_5 : if\, f_2\, and\, f_3,\, then\, g_2.$

Figure 1 provides an example of AND/OR graph, where:

$s_1, s_2, s_3, s_4, s_5 \in S; f_1, f_2, f_3 \in F; r_1, r_2, r_3, r_4, r_5 \in R; g_1, g_2 \in G.$

Rule r_i as

$$r_{i:}\ \text{if}\, f_{r_i 1}\ \text{and}\, f_{r_i 2} \ldots f_{r_i n},\ \text{then}\, f_{r_i m}$$

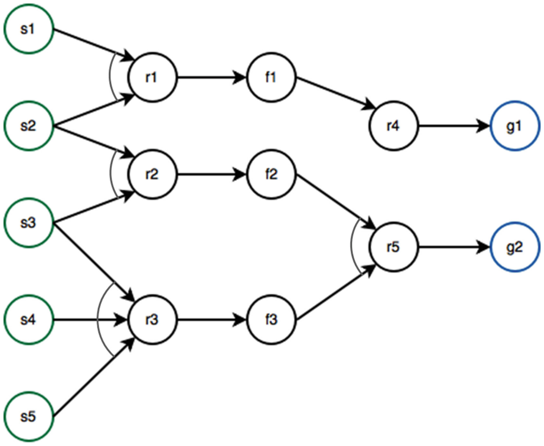


Fig. 1. AND/OR graph example

can be represented as a pair $r_i = (D_{r_i}; Q_{r_i})$, where $D_{r_i} = \{f_{r_i}1, f_{r_i}2, ..f_{r_i}n\}$ and $Q_{r_i} = \{f_{r_i}m\}$. Q_{r_i} has a single element, henceforward defined as q_{r_i}.

Let L is a set of inference chains.

Definition 1. An inference chain l_i – is a sequence of rules $(r_{l_i}1, r_{l_i}2, \ldots, r_{l_i}n)$, if $\forall r_{l_i k}, r_{l_i(k+1)}, q_{r_{l_i k}} \in D_{r_{l_i(k+1)}}$, where $k = 2, \ldots, (n-1)$.

Then, the graph on Fig. 1 has $L = \{l_1, l_2, l_3, l_4, l_5, l_6, l_7, l_8\}$, where

$$l_1 = (r_1); l_2 = (r_2); l_3 = (r_3); l_4 = (r_4); l_5 = (r_5); l_6 = (r_{1,}r_4);$$
$$l_7 = (r_2, r_5); l_8 = (r_3, r_5).$$

Definition 2. The start of inference chain l_i as $(r_{l_i}1, r_{l_i}2, \ldots, r_{l_i}n)$, is a set of the facts in condition of the first rule of the chain, $D_{l_i} = D_{r_{l_i}1}$.

Definition 3. The end of inference chain l_i as $(r_{l_i}1, r_{l_i}2, \ldots, r_{l_i}n)$ is a result of the last rule of the chain, $Q_{l_i} = Q_{r_{l_i n}}$.

Definition 4. Structural error in the rule-based system is an error which can be detected during AND/OR graph analysis. Knowledge base which doesn't have any structural errors is considered as a statically correct one.

The classification of structural errors is provided in Fig. 2.

2.1 Redundancy Errors

2.1.1 Duplicates

Definition 5. Rules r_i and r_j are considered as duplicates, if $D_{r_i} \cap D_{r_j} \neq \emptyset$ and $q_{r_i} = q_{r_j}$.

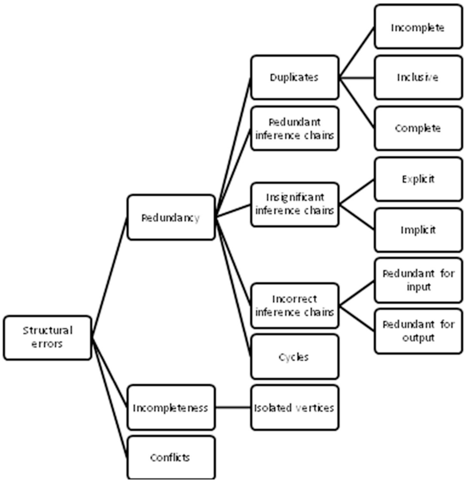


Fig. 2. Structural errors classification

There are three types of duplicates:

- inclusive duplicates;
- complete duplicates;
- incomplete duplicates.

Definition 6. Rules r_i and r_j are considered as inclusive duplicates, if $D_{r_\mathrm{i}} \subset D_{r_j} . D_{r_\mathrm{i}} \; / = D_{r_\mathrm{j}}$ and $q_{r_\mathrm{i}} = q_{r_j}$, r_i is defined as included one in this case.

Rules r_1 and r_2 in the Fig. 3. are inclusive duplicates:

$$r_1 : \mathit{if}\, f_1\, \mathit{and} f_2,\, \mathit{then} f_4;$$
$$r_2 : \mathit{if}\, f_1\, \mathit{and} f_2\, \mathit{and}\, f_3,\, \mathit{then} f_4;$$
$$D_{r_1} = \{f_1, f_2\}; \quad D_{r_2} = \{f_1, f_2, f_3\}; \quad D_{r_1} \,\&\, D_{r_2}$$
$$q_{r_1} = f_4; \quad q_{r_2} = f_4;$$

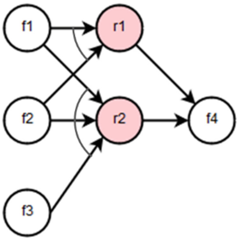


Fig. 3. Inclusive duplicates

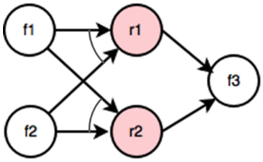


Fig. 4. Complete duplicates

The solution to correct inclusive duplicates error is to remove all duplicating rules except included one.

Definition 7. Rules r_i and r_j are considered as complete duplicates, if $D_{r_\mathrm{i}} = D_{r_\mathrm{j}}$ and $q_{r_\mathrm{i}} = q_{r_j}$.

Rules *r1* and *r2* in the Fig. 4. are complete duplicates:

$r_1 : if\, f_1\, and f_2,\ then f_3;$
$r_2 : if\, f_1\, and f_2,\ then f_3;$

The solution to correct complete duplicates error is to remove all duplicating rules except one.

Definition 8. Rules r_i and r_j are considered as incomplete duplicates, if $D_{r_\mathrm{i}} \cap D_{r_j} \neq \emptyset$, $q_{r_\mathrm{i}} = q_{r_\mathrm{j}}$, $D_{r_\mathrm{i}} \backslash D_{r_j} \neq \emptyset$ and $D_{r_\mathrm{j}} \backslash D_{r_\mathrm{i}} \neq \emptyset$.

The Fig. 5. provides an example of incomplete duplicates:

$r_1 : if\, f_1\, and f_2,\ then f_4;$
$r_2 : if\, f_2\, and f_3,\ then f_4;$
$D_{l_1} \cap D_{l_2} = f_2;$

There is no single solution for correcting incomplete duplicates errors, so the concrete decision on the way of the improvement of the KB and correctness the error should be made by the expert in each particular case.

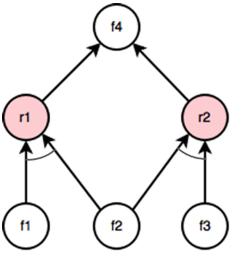


Fig. 5. Incomplete duplicates

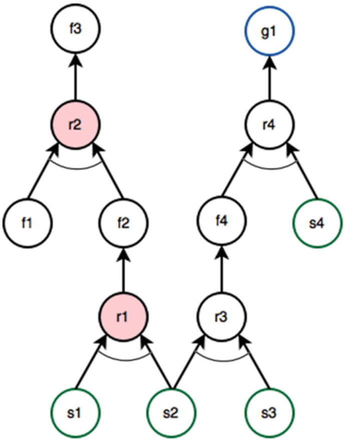


Fig. 6. Redundant inference chain

2.1.2 Redundant Inference Chains

Definition 9. An inference chain l_i is as considered redundant, if $q_{l_i} \notin G$ and $\neg \exists l_j$, where $q_{l_i} \in D_{l_j}$ and $q_{l_j} \in G$.

The Fig. 6 provides an example of redundant inference chain $-l_1 = (r_1, r_2)$, where:

r_1 : *if* s_1 *and* s_2, *then* f_2;
r_2 : *if* f_1 *and* f_2, *then* f_3;
$q_{l_1} = f_3; f_3 \notin G$;

Redundant inference chains can be removed from a knowledge base.

2.1.3 Insignificant Inference Chains

Definition 10. An inference chain l_i as $(r_{l_i1}, r_{l_i2}, \ldots, r_{l_in})$ is considered as insignificant one, if $\forall r_{l_ij}, \left|D_{r_{l_ij}}\right| = 1$, where $j = 1, \ldots, n$.

$l_1 = (r_1)$ in the Fig. 7. is an insignificant inference chain:

$D_{r_1} = \{f_1\}; \quad |D_{r_1}| = 1;$

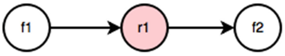


Fig. 7. Insignificant inference chain

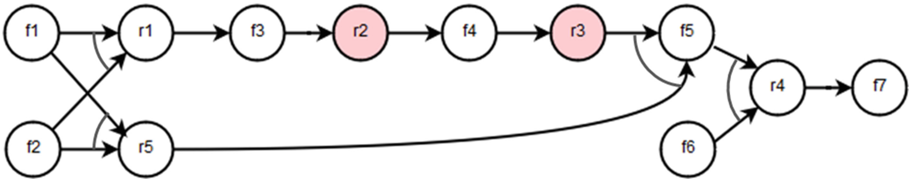


Fig. 8. Explicit insignificant inference chain

There are two types of insignificant inference chains:

- explicit chains;
- implicit chains.

Definition 11. An insignificant inference chain l_i is as considered explicit one, if $\exists r_j$, where $r_j = (D_{l_i}; Q_{l_i})$.

The Fig. 8 provides an example of explicit insignificant inference chain $l_1 = (r_2, r_3)$, where:

r_2 : *if* f_3, *then* f_4;
r_3 : *if* f_4, *then* f_5;
$D_{l_1} = \{f_3\}; q_{l_1} = \{f_5\}$;
And the rule $r_5 = \{D_{l_1}; q_{l_1}\}$ *exists.*

The solution is to remove explicit insignificant inference chain.

Definition 12. An insignificant inference chain l_i is as considered implicit one, if $\neg\exists r_j$ where $r_j = (D_{l_i}; Q_{l_i})$.

The Fig. 9. provides an example of implicit insignificant inference chain $l_1 = (r_2, r_3)$, where:

r_2 : *if* f_3, *then* f_4;
r_3 : *if* f_4, *then* f_5;
$D_{l_1} = \{f_3\}; q_{l_1} = \{f_5\}; r_5 = \{D_{l_1}; q_{l_1}\}$
And the rule $r_k = \{D_{l_1}; q_{l_1}\}$ *doesn't exist.*

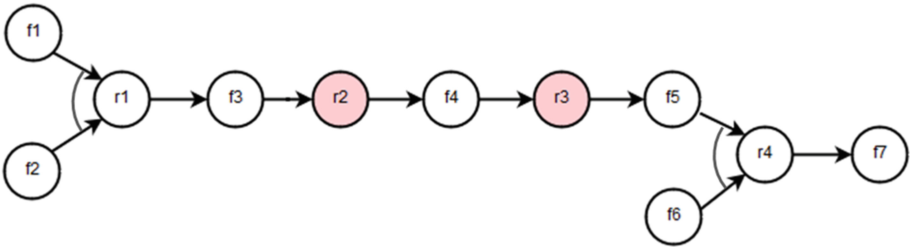


Fig. 9. Implicit insignificant inference chain

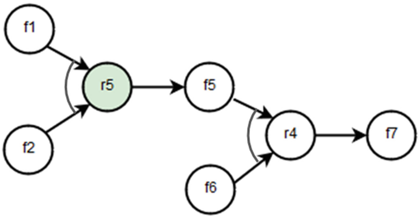


Fig. 10. Transforming to the preceding rule

An implicit insignificant inference chain is not a critical one for the knowledge base, but it allows optimization by transforming to the preceding or succeeding rule.

Definition 13. Transforming to the preceding rule r_n of the insignificant inference chain l_i where $q_{r_\mathrm{n}} \in D_{l_\mathrm{i}}$, is a creation of a rule r_m, where $r_m = (D_{r_n}; Q_{l_i})$, and deletion of l_i and r_n from the knowledge base.

The transformation to the preceding rule of the insignificant inference chain in the Fig. 9 is shown in the Fig. 10. A new rule r_5 has been added:

r_5 : *if* f_1 *and* f_2, *then* f_5;

Definition 14. Transforming to the succeeding rule r_n of the insignificant inference chain l_i where $q_{l_\mathrm{i}} \in D_{r_\mathrm{n}}$, is a creation of the rule r_m, where $r_m = ((D_{r_n} - q_{l_i}) \cup D_{l_i};\ Q_{r_n})$, and deletion of l_i and r_n from the knowledge base.

The transformation to the succeeding rule of the insignificant inference chain in the Fig. 9 is shown in the Fig. 11. A new rule r_5 has been added:

r_5 : *if* f_3 *and* f_6, *then* f_7;

2.1.4 Incorrect Inference Chains

There are two types of incorrect inference chains:

- redundant for input;
- redundant for output.

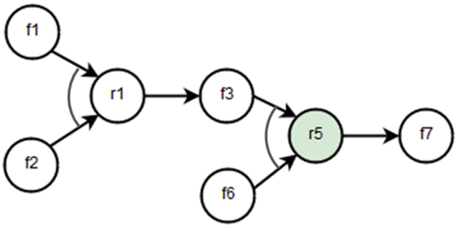


Fig. 11. Transforming to the succeeding rule

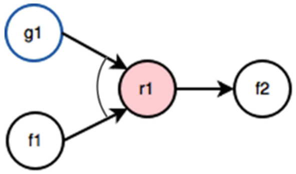


Fig. 12. Inference chain redundant for input

Definition 15. An inference chain $\mathrm{l_i}$ is considered as redundant for input one, if $D_{l_{\mathrm{i}}} \cap G \neq \emptyset$.

The Fig. 12 provides an example of a redundant for input inference chain $l_1 = (r_1)$, where:

$r_1 : \mathit{if}\, g_1\, \mathit{and} f_1,\, \mathit{then} f_2;\, g_1 \in G$

The solution is to remove the inference chain redundant for input.

Definition 16. An inference chain l_i is considered as redundant for output one, if $q_{l_{\mathrm{i}}} \in S$.

The Fig. 13. provides an example of a redundant for output inference chain $l_1 = (r_1)$, where:

$r_1 : \mathit{if} f_1\, \mathit{and} f_2,\, \mathit{then}\, s_1; s_1 \in S.$

The solution is to remove the inference chain redundant for output one.

2.1.5 Cycles

Definition 17. An inference chain l_i is considered as a cycle, if $q_{l_{\mathrm{i}}} \in D_{l_{\mathrm{i}}}$.

The Fig. 14. provides an example of a cycle $l_1 = (r_1, r_2, r_3)$, where:

$r_1 : \mathit{if} f_1\, \mathit{and} f_2,\, \mathit{then} f_3;$
$r_2 : \mathit{if} f_3\, \mathit{and} f_4,\, \mathit{then} f_5;$
$r_3 : \mathit{if} f_5\, \mathit{and} f_6,\, \mathit{then} f_2;$
$D_{l_1} = \{f_1, f_2\};\; q_{l_1} = \{f_2\};\; q_{l_1} \in D_{l_1};$

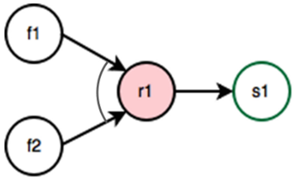


Fig. 13. Inference chain redundant for output

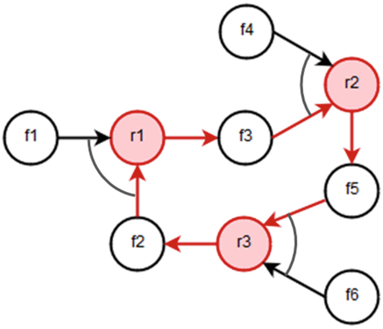


Fig. 14. Cycle

Any cycle discovered in the knowledge base is a structural error, but incorrect rule cannot be selected automatically, so an expert should choose what rule to be removed from the KB.

Definition 18. A rule r_i is considered as a simple cycle, if $q_{r_{\mathrm{i}}} \in D_{r_{\mathrm{i}}}$

The rule r_1 in the Fig. 15 provides an example of a simple cycle:

$r_1 : if\, f_1\, and\, f_2,\, then\, f_1;$

A simple cycle should be removed from the knowledge base.

2.2 Incompleteness Errors

2.2.1 Isolated Vertices

Definition 19. A vertex $f_i \in F \cup G$ is considered isolated, if $\neg \exists r_j$, where $f_i \in D_{r_{\mathrm{j}}}$ or $f_i \in Q_{r_{\mathrm{j}}}$.

The vertex g_1 in the Fig. 16 is an isolated one, because there is no rule connected to it.

The correction depends on type of the vertex. If the isolated vertex is an input fact or a goal, the solution is to add rules by the expert. Otherwise the vertex should be removed from the knowledge base.

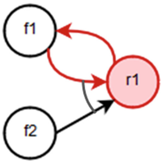


Fig. 15. Simple cycle

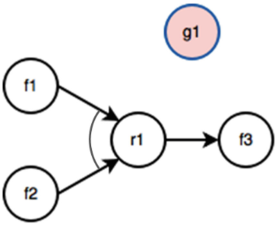


Fig. 16. Isolated vertex

2.3 Inconsistency

C is a set of allowed combinations of facts, so $c_i = \{f_{c_i 1}, f_{c_i 2}, \ldots, f_{c_i n}\}$.

2.3.1 Conflicting Inference Chains

Definition 20. Inference chains $\mathrm{l_i}$ and $\mathrm{l_j}$ are considered as conflicting ones, if $\exists\, f_k$, where $f_k \in D_{l_i}$ and $f_k \in D_{l_j}$, and $\neg\exists c_m$, where $q_{l_i} \in c_m$ and $q_{l_j} \in c_m$.

There are inference chains $l_1 = (r_1,\, r_2)$ and $l_2 = (r_3)$ in the Fig. 17:

$$D_{l_1} = \{f_1, f_2\}; q_{l_1} = \{f_7\}$$
$$D_{l_2} = \{f_2, f_3\}; q_{l_2} = \{f_6\};$$

There is no $c_k = (f_6, f_7)$ and $D_{l_1} \cap D_{l_2} = f_2$, so l_1 and l_2 are considered conflicting inference chains.

The correction of conflicting chains error should be done by the expert, either by adding a new allowed combination of facts, or by rewriting or deleting rules.

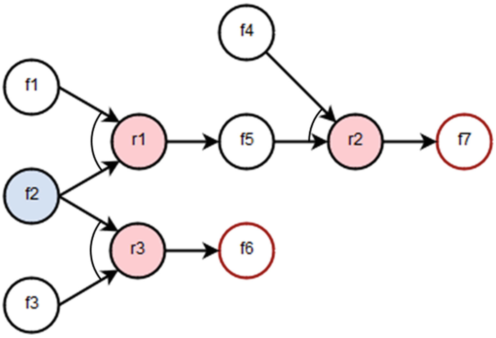


Fig. 17. Conflicting inference chains

3 Conclusion

The paper provides formal classification of structural errors in the knowledge bases of intellectual decision making systems. It describes 3 types of structural errors: redundancy errors, incompleteness errors and inconsistency. For each of group examples are provided. Duplicates, redundant inference chains, insignificant inference chains, incorrect inference chains and cycles are considered as redundancy errors, and isolated vertices error type is incompleteness. Inconsistency in knowledge bases is represented by conflicting inference chains. All errors are formalized in terms of the AND/OR graph model and ways of correction are provided for each defined structural error.

The formal model of structural errors can be used as a basis for performing of automatic verification of rule-based knowledge bases structure.

References

1. Dolinina, O.N.: Razrabotka metoda testirovanija produkcionnyh baz znanij jekspertnyh sistem s uchetom oshibok tipa «zabyvanija ob iskljuchenii»: dis…kand. tehn. nauk/O.N. Dolinina. Saratov, 171 s (1999) (in Russian)
2. Dolinina, O.: Method of the debugging of the knowledge bases of intellectual decision making systems. In: Silhavy, R., Senkerik, R., Oplatkova, Z.K., Silhavy, P., Prokopova, Z. (eds.) Automation Control Theory Perspectives in Intelligent Systems. AISC, vol. 466, pp. 307–314. Springer, Cham (2016). doi:10.1007/978-3-319-33389-2_29
3. Nguen, T., Perkins, W., Laffey, T., Pecors, W.: Checking expert system knowledge bases for consistency and completeness. In: Proceedings of the 9th International Joint Conference on Artificial Intelligence, Los Angeles, pp. 375–378, August 1985
4. Cragun, B.J., Stendel, H.J.: A decision-table-based processor for checking completeness and consistency in rule-based expert systems. Int. J. Man-Mach. Stud. **26**(5), 633–648 (1987)
5. Nguen, T.A.: Verifying consistency of production systems. In: Proceedings of the Third Conference on Artificial Intelligence Applications (CAIA), Kissimmee, Fl, pp. 4–8 (1987)
6. Suwa, M., Scott, A.C., Shortliffe, E.H.: An approach to veryfing consistency and completeness in a rule-based expert system. In: Rule-Based Expert Systems, pp. 159–170. Addison-Wesley, London (1984)
7. Nguen, T., Perkins, W., Laffey, T., Pecora, W.: Checking expert system knowledge bases for consistency and completeness. In: Proceedings of the 9th International Joint Conference on Artificial Intelligence, Los Angeles, pp. 375–378, August 1985

Hybrid Particle Swarm Optimization and Backpropagation Neural Network for Organic and Inorganic Waste Recognition

Christ R.A. Djaya, Novi Sucianti, Randy, and Lili A. Wulandhari(✉)

School of Computer Science, Bina Nusantara University,
Jl. K.H. Syahdan No. 9, 11480 Jakarta, Indonesia
christflvy@gmail.com, suciantinovi@gmail.com,
randygun24@gmail.com, lwulandhari@binus.edu

Abstract. Separation of organic and inorganic waste for daily need is one of efforts to yield sanitation. However, most people have difficulties to distinguish these kind of waste. Therefore this paper propose a system that can recognize organic and inorganic waste automatically. These system is developed using hybrid PSO-BPNN algorithm to recognize type of waste. Input data is organic and inorganic image which is captured around campus. This paper also presents comparison of BPNN, PSO and PSO-BPNN in recognizing type of waste. The results show that each algorithm achieves 77%, 69% and 95% for BPNN, PSO and hybrid PSO-BPNN respectively.

Keywords: Waste recognition · Organic · Inorganic · Hybrid PSO-BPNN · PSO · BPNN

1 Introduction

Waste is one of the common problem that government has been faced from time to time in every city in Indonesia, especially Jakarta. In 2003, Jakarta has produced 5000 tons of waste in a day from their citizen which is approximately 10 million. Even, Indonesia has reached 100.000 tons in a day and it increases every year by 4% [1], and it requires government concern to overcome this problem. It makes the government must work hard to overcome this problem.

Waste is divided into organic and inorganic based on content and degradation level. Organic waste contains organic compound and easy to be degraded like leaf litter and leftovers. While, inorganic waste is the reverse of organic waste, it does not contain organic compound and hard to be degraded like plastics and cans [2]. Organic and inorganic waste should be disposed and placed in different places, because some of the waste may contain any poisonous material which can harm the environment. However, in some place organic and inorganic waste separation has not been handled properly because it is often found waste are mixed in the same bin. It happens because people still have difficulty in determining whether a waste is organic or inorganic. Therefore, it is required a tool that can assist people to separate organic and inorganic waste automatically.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_17

Previous researchers had been proposed some techniques in waste separation. Salmador et al. (2008) proposed K-Nearest Neighbor (KNN) to separate household and industrial waste which is applied in robot. Another researcher proposed mechanical approach to sort solid waste such as metal, plastic, glass, paper and old tiers to be recycled [3]. Intelligent system also be applied by previous researcher to separate inorganic such as aluminum cans, plastic bottles and plastic cutleries. They applied computer vision approach and KNN to identify type of wastes [4]. Waste separation especially in Indonesia, needs a system to recognize organic and inorganic automatically. Based on previous researches, it is found that researchers only proposed technique to recognized inorganic waste. Therefore, this paper presents a technique to identify organic and inorganic by using computational intelligence especially Backpropagation Neural Network (BPNN) and Particle Swarm Optimization (PSO).

BPNN is one of the most popular technique in recognition [5–8] which is implemented in machine, health, voice recognition and electrical field. BPNN is a suitable approach for modeling behavior of a system even if the data are affected by noise [9, 10]. However, BPNN encounters drawback due to random initial weight which affects to local minimum and bad convergent velocity [11]. Therefore, it is important to optimize initial weight of BPNN to overcome this weakness. PSO is one of the optimization technique that can be used in initial weight optimization.

PSO is a recognition algorithm that can find a set of unknown parameter better [12]. Compared to Genetic Algorithm (GA), PSO has a better performance since PSO has the ability to do global and local search simultaneously [13]. Moreover, PSO has some advantages like easy to be implemented and more flexible in maintaining the balance of global and local search from their search space [14].

This paper presents a technique to recognize organic and inorganic waste by using waste images as the input. Computer vision approach is implemented to process raw image data to obtain features which can be used as input in hybrid PSO-BPNN. This paper is arranged as follows; Section 1 explains problem background and current research in waste identification, Sect. 2 discusses about data collection and analysis, Sect. 3 describes proposed methodology which follows with result and discussion in Sect. 4. Finally, conclusion will be represented in Sect. 5.

2 Image Collection and Preprocessing

This research uses organic and inorganic waste images as the data which is usually found around college. Total data collected are 40 images namely 20 organics and 20 inorganics image with dimension 960 × 700. Organic data consist of paper, tissue, leaf and box of milk (see Fig. 1) where inorganic data consist of plastic, can and bottle (see Fig. 2). Each image will pass preprocessing for instances; grayscaling, edge detection and feature extraction before used as input data.

First step of image pre-processing is Grayscaling, this process will change the image which contain 3 layers of color to grayscale image which contains only one layer of color (see Fig. 3). The formula to convert color image to grayscale image is given as follows [15]:

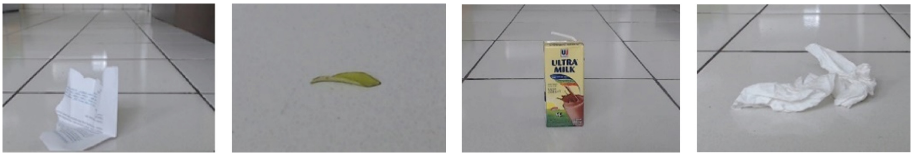


Fig. 1. Organic waste

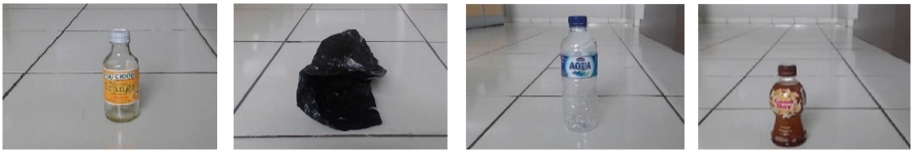
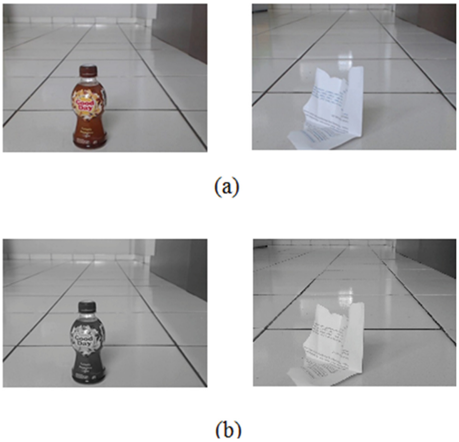


Fig. 2. Inorganic waste



(a)

(b)

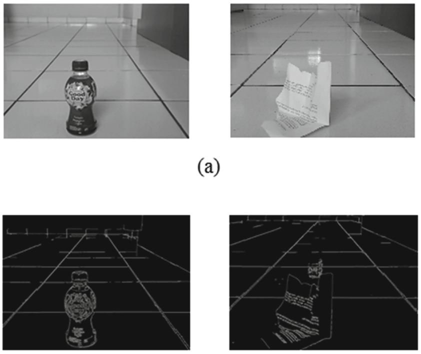
Fig. 3. Image before grayscaling (a), image after grayscaling (b)

$$I_g(i,j) = \frac{R(i,j) + G(i,j) + B(i,j)}{3} \tag{1}$$

Where

I_g = Intensity value of grayscale image at i,j pixel
R = Intensity value of red color on image at i,j pixel
G = Intensity value of green color on image at i,j pixel
B = Intensity value of blue color on image at i,j pixel

Grayscale image that we got from grayscaling process will through an edge detection process to detect each edge from the image object because training and



(a)

(b)

Fig. 4. Grayscale image (a) binary image after canny edge detection (b)

recognizing will be done based on object shape at the image. Canny Edge Detection algorithm is applied to obtain edge of image (see Fig. 4). This algorithm is well known for its performance in handling noise [16]. Process of canny edge detection is divided into five steps as follows:

Smoothing. Smoothing is a process of softening image to reduce the noise, usually using Gaussian Filter. Every image that taken from camera will have some noise and to avoid incorrect detection of noise into the edge, smoothing must be done.

Finding Gradient. Usually, Canny detect edge point on grayscale which had highest intensity value changes. Those areas discovered by determining gradient of an image which is calculated using Gradient Magnitude formula:

$$G = \sqrt{G_x^2 + G_y^2} \tag{2}$$

Where

G = Gradient magnitude
G_x = Gradient of horizontal direction (x)
G_y = Gradient of vertical direction (y)

As for calculating the angle direction will use the following formula:

$$\theta = \arctan\left(\frac{G_y}{G_x}\right) \tag{3}$$

Where

θ = Gradient of angle direction
G_x = Gradient of horizontal direction (x)
G_y = Gradient of vertical direction (y)

Angle produced from the calculation will be rounded to 0, 45, 90 and 135° based on closure.

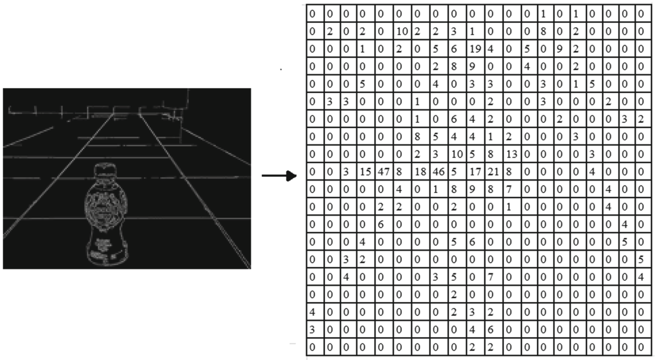
Non Maximum Suppression The next step is get rid of non-maximum values because only local maximum that will be used as edge. Other pixel which have value that are not local maximum will be lowered to zero.

Double Thresholding. The remaining edge pixels from the previous step will be marked as strong pixels. Most of them is the real edge on image, but some may be caused by noise or color variations by rough surfaces. So, to distinguish them is by using a threshold value so that only the edge with a strong value will be kept. Canny algorithm uses double thresholding system (upper threshold and lower threshold) where the edges with a value higher than the upper threshold marked as strong point. Edges with a value less than lower threshold will be deleted and edges with value between upper threshold and lower threshold will be marked as weak point and will be deleted, except it is connected to edge which have strong point.

Edge Tracking by Hysteresis. To know the relation between each edge, edge tracking done using Binary Large Object. Binary Large Object applied by looking at weak point and the 8-Neighborhood. As long as there is a strong pixel on Binary Large Object, the weak pixel could be identified as something that can be included or stored.

2.1 Feature Extraction

Feature extraction is required to reduce number of neuron which be used in recognition process. Original binary image has 960 × 700 dimension which will represent as neuron in input layer. Feature extraction is carried out by segmented the original image into 48 × 35 dimension and calculate number of element 1 in each segmentation. This segmentation produces 20 × 20 features which be represented into 400 input neurons for every image (see Fig. 5).



0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0
0	2	0	2	0	10	2	2	3	1	0	0	0	8	0	2	0	0	0	0
0	0	0	1	0	2	0	5	6	19	4	0	5	0	9	2	0	0	0	0
0	0	0	0	0	0	0	2	8	9	0	0	4	0	0	2	0	0	0	0
0	0	0	5	0	0	0	4	0	3	3	0	0	3	0	1	5	0	0	0
0	3	3	0	0	0	1	0	0	0	2	0	0	3	0	0	0	2	0	0
0	0	0	0	0	0	1	0	6	4	2	0	0	0	2	0	0	0	3	2
0	0	0	0	0	0	8	5	4	4	1	2	0	0	0	3	0	0	0	0
0	0	0	0	0	0	2	3	10	5	8	13	0	0	0	0	3	0	0	0
0	0	3	15	47	8	18	46	5	17	21	8	0	0	0	0	4	0	0	0
0	0	0	0	0	4	0	1	8	9	8	7	0	0	0	0	0	4	0	0
0	0	0	0	2	2	0	0	2	0	0	1	0	0	0	0	0	4	0	0
0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0
0	0	0	4	0	0	0	0	5	6	0	0	0	0	0	0	0	0	5	0
0	0	3	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5
0	0	4	0	0	0	0	3	5	0	7	0	0	0	0	0	0	0	0	4
0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	2	3	2	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	4	6	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	2	2	0	0	0	0	0	0	0	0	0

(a) (b)

Fig. 5. Binary image (a) extraction feature result (b)

3 Organic-Inorganic Recognition Technique

Organic-inorganic identification algorithm uses the combination of BPNN and PSO as optimization algorithm. PSO will take part in optimizing weight and bias value where every set of weight and bias will be used as particles in PSO. After PSO performed, the optimized set of weight and bias will be used for BPNN. The scheme of PSO-BPNN is given in a Fig. 6.

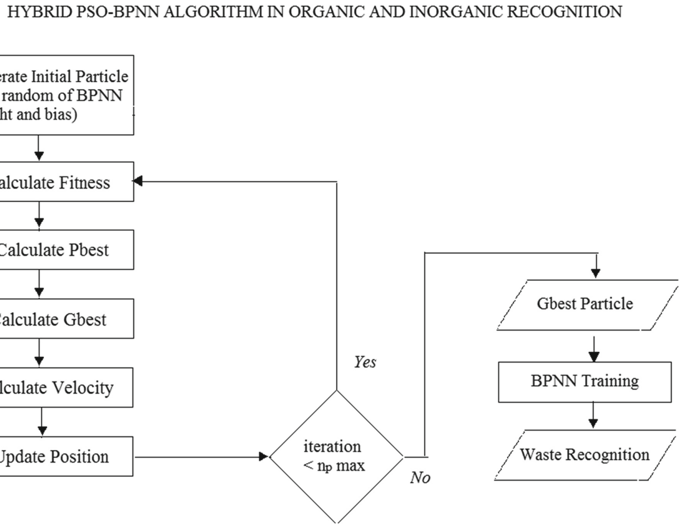


Fig. 6. Hybrid PSO-BPNN flowchart

Weight and bias values are taken a randomly in range $[-2, 2]$ that will be used as initial position of particle. Fitness function of PSO is obtained from Mean Square Error (MSE) with output O is obtained from forward process at BPNN which apply binary sigmoid as activation function.

$$O = \frac{1}{1 + e^{-(\sum w \cdot I + b)}} \tag{4}$$

Where w and b is set of weights and bias respectively. The algorithm of PSO-BPNN is described in the following steps:

1. Specify the topology of the BPNN to determine number of initial weights as the initial swarm of particle.
2. Let (I_b, T_b) is the bth input and target pair for BPNN, with $b = 1, 2, \ldots, N_{data}$ and N_{data} is the number of paired data.
3. Let N_i, N_x and N_{it} is the number of particle, number of dimension and number of iterations respectively, and $t_{it\,max}$ is maximum iteration.

4. Generate an initial particle. Each particle has dimension which correspond to BPNN random weights.
5. Set $t = 0$

   While $t \leq t_{it\,max}$ do

   Calculate fitness value $F(x_i)$ of the ith particle in swarm S_i.

$$F(x_i) = E(x_i, I_b, T_b) \qquad i = 1, 2, \ldots, N_i \tag{5}$$

   Where $E(x_i, I_b, T_b)$ is the Mean Square Error (MSE) of the ith particle using (I_b, T_b) pair of data. It is calculated based on the selected BPNN architecture and activation function;

$$E(x_i, I_b, T_b) = \frac{1}{2}\sum_{b=1}^{N_{data}} \left(T_b - O_{b(x_i)}\right)^2 \tag{6}$$

   Where T_b is Target of the bth input and $O_{b(x_i)}$ is Output of the bth input which obtain using ith particle based on the selected BPNN architecture.

   Let x_i denote position of particle ith and y_i denote the best position of particle ith

```
For each particle $i = 1,2,\ldots,N_i$ do
    If $F(x_i) < F(y_i)$ then
        $y_i = x_i$
    End
```

```
Let $\hat{y}$ denote the global best position in a swarm
    If $F(y_i) < F(\hat{y})$ then
        $\hat{y} = y_i$
    End
```

```
For each particle $i = 1,2,\ldots,N_i$ do
    Update the velocity using equation:
```

$$v_{ij}(t+1) = wv_{ij}(t) + c_1 r_{1j}(t)\left[y_{ij}(t) - x_{ij}(t)\right] + c_2 r_{2j}(t)\left[\hat{y}_j(t) - x_{ij}(t)\right] \tag{7}$$

   Where $v_{ij}(t)$ is the velocity of ith particle in dimension $j = 1, 2, \ldots, N_x$ at iteration t. $x_{ij}(t)$ is the position of ith particle in dimension $j = 1, 2, \ldots, N_x$ at iteration t. w is inertia weight, where this research uses linear decreasing inertia weight [17] c_1 and c_2 are positive constant acceleration, $r_{1j}(t)$ and $r_{2j}(t)$ are random values in range $[0, 1]$.

   Update the position using equation:

$$x_{ij}(t+1) = v_{ij}(t) + v_{ij}(t+1) \tag{8}$$

   With $x_{ij}(0)$ in range (x_{min}, x_{max}).

$$t = t + 1$$

   End
6. Apply the best solution in current swarm as the initial weights for BPNN learning.

The BPNN learning from the BPNN part produces recognition of the organic and inorganic waste where its performance is validated by calculating MSE and percentage of accuracy. The performance results of PSO-BPNN algorithms are presented in Sect. 4.

4 Result and Analysis

This section presents performance evaluation of PSO-BPNN and its comparison with individual BPNN and individual PSO in MSE and percentage of accuracy. Percentage of accuracy is defined as follows [18]:

$$Accuracy(\%) = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \tag{9}$$

Assuming percentage of accuracy above is corresponds to the P class, where TP is true positives, which means that the prediction gives correct classification of P class according to the actual; *TN* is true negatives, which means the prediction gives correct label classification for all non-P class;*FN* is false negatives, which means the prediction gives incorrect classification label of the actual P class and *FP* is false positives, which means the prediction gives incorrect classification labels another class as P class.

Performance evaluation of PSO-BPNN is conducted by using 400-400-1 BPNN architecture, namely 400 neurons in input layer, 400 neurons in hidden layer and 1 neuron in output layer. The data is divided into 80 data for training and 20 data for testing. Training is conducted 5 times for 50 and 100 iterations for each algorithm with learning rate 0.5 for BPNN, 0.3 and 0.5 for c_1 and c_2 respectively for PSO. The comparison of BPNN, PSP and PSO-BPNN performance is given in Table 1 as follows.

From training result with 50 iterations, it is known that BPNN has MSE 0.051201 and 11.56 s for processing time. While PSO has 0.120857of average error and 49.31 s of average processing time which is 0.069 higher compare with BPNN. Hybrid PSO-BPNN has the lowest MSE with difference 0.04 with BPNN. While for training with 100 iterations, it is obtain that hybrid PSO-BPNN also has the lowest MSE compared with BPNN and PSO. Testing results for 50 and 100 iterations, also shows that Hybrid PSO-BPNN has the highest accuracy and achieve around 22% and 17% higher compare with BPNN for 50 and 100 iteration respectively.

Table 1. Training result

Iteration	Algorithm	Average error	Processing time (s)	Testing accuracy
50	BPNN	0.051201	11.56	70%
	PSO	0.120857	49.31	62%
	Hybrid PSO-BPNN	**0.003760**	**13.87**	**92%**
100	BPNN	0.053751	22.68	77%
	PSO	0.120293	110.77	69%
	Hybrid PSO-BPNN	**0.003751**	**25.26**	**95%**

5 Conclusion

This paper presents hybrid PSO-BPNN for organic and inorganic recognition. The experiments result show that PSO is capable to optimize weights of BPNN, such that BPNN learning can achieve good result. From the evaluation PSO-BPNN has higher accuracy compare to individual BPNN with only 2.58 s for additional processing time. This results shows that PSO-BPNN has good performance and can be implemented to develop organic-inorganic separation system.

Acknowledgments. The authors thank to School of Computer Science (SoCS) and Research Technological Transfer Office (RTTO) of Bina Nusantara University, Indonesia for supporting this research.

References

1. Hartono, R.: Penanganan dan Pengolahan Sampah. Penebar Swadaya, Jakarta (2008)
2. Sejati, K.: Pengolahan Sampah Terpadu dengan Sistem Node, Sub Point dan Center Point. Kanisius, Yogyakarta (2009)
3. Huang, J., Pretz, T.. Bian, Z.: Intelligent solid waste processing using optical sensor based sorting technology. In: International Congress on Image and Signal Processing (CISP 2010), pp. 1657–1661 (2010)
4. Torres-García, A., Rodea-Aragón, O., Longoria-Gandara, O., Sánchez-García, F., González-Jiménez, L.E.: Intelligent waste separator. Computación y Sistemas **19**(3), 487–500 (2015)
5. Payganeh, G., Khajavi, M.N., Ebrahimpour, R., Babaei, E.: Machine fault diagnosis using MLPs and RBF neural networks. In: Applied Mechanics and Materials, pp. 5021–5028 (2012)
6. Zhao, K., Wang, C., Hu, J., Yang, X., Wang, H., Li, F., Zhang, X., Zhang, J., Wang, X.: Prostate cancer identification: quantitative analysis of T2-weighted MR images based on a back propagation artificial neural network model. Sci. China Life Sci. **58**, 666–673 (2015)
7. Hosom, J.-P., Vermeulen, P.J., Shaw, J.: Speaker verification and identification using artificial neural network-based sub-phonetic unit discrimination. United States Patent US 9230550 B2 (2016)
8. Roman, A.J., Kreitzer, P.J., Ervin, J.S., Hanchak, M.S., Byyd, L.W.: Flow pattern identification of horizontal two-phase refrigerant flow using neural networks. Int. Commun. Heat Mass Transf. **71**, 254–264 (2016)
9. Puscasu, G., Palade, V., Stancu, A., Buduleanu, S., Nastase, G.: Sisteme de Conducere Clasice si Inteligente a Proceselor. MATRIX ROM, Bucharest (2000)
10. Bocaniala, C.D., Palade, V.: Computational intelligence methodologies in fault diagnosis: review and state of the art. In: Palade, V., Jain, L., Bocaniala, C.D. (eds.) Advanced Information and Knowledge Processing, pp. 1–36. Springer, London (2006)
11. Nawi, N.M., Khan, Abdullah, Rehman, M.Z.: A new back-propagation neural network optimized with cuckoo search algorithm. In: Murgante, B., et al. (eds.) ICCSA 2013. LNCS, vol. 7971, pp. 413–426. Springer, Heidelberg (2013). doi:10.1007/978-3-642-39637-3_33
12. Zhao, H.-B., Yin, S.: Geomechanical parameters identification by particle swarm optimization and support vector machine. Appl. Math. Modell. **33**(10), 3997–4012 (2009)

13. Rajendra, R., Pratihar, D.K.: Particle swarm optimization algorithm vs genetic algorithm to develop integrated scheme for obtaining optimal mechanic structure and adaptive controller of a robot. Intell. Control Autom. **2**(4), 430–449 (2011)
14. Sathya, P.D., Kayalvizhi, R.: PSO-based Tsallis thresholding selection procedure for image segmentation. Int. J. Comput. Appl. **5**(4), 39–46 (2010)
15. Kanan, C., Cottrell, G.W.: Color-to-grayscale: does the method matter in image recognition? Plos One **7**(1), e29740 (2012)
16. Shrivakshan, G.T., Chandrasekar, C.: A comparison of various edge detection techniques used in image processing. IJCSI Int. J. Comput. Sci. Issues **9**(5), 272–276 (2012)
17. Bansal, J.C., Singh, P.K., Saraswat, M., Verma, A., Jadon, S.S., Abraham, A.: Inertia weight strategies in particle swarm optimization. dalam: 2011 Third World Congress on Nature and Biologically Inspired Computing (2011)
18. Kohavi, R., Provost, F.: Glossary of terms: special issue on applications of machine learning and the knowledge discovery proces. Mach. Learn. **30**, 271–274 (1998)
19. Salmador, A., Cid, J.P., Novelle, I.R.: Intelligent garbage classifier. Int. J. Interact. Multimed. Artif. Intell. **1**(1), 31–36 (2008)

Calibration of Low-Cost Three Axis Accelerometer with Differential Evolution

Ales Kuncar(✉), Martin Sysel, and Tomas Urbanek

Faculty of Applied Informatics, Tomas Bata University in Zlin,
Namesti T.G. Masaryka 5555, 76001 Zlin, Czech Republic
{kuncar,sysel,turbanek}@fai.utb.cz

Abstract. The accelerometers are used in wide range of engineering applications. However, the accuracy of accelerometer readings is influenced by many factors such as sensor errors (scale factors, non-orthogonality, and offsets); therefore, the accelerometer calibration is necessary before its use in advanced applications. This research paper describes calibration methods for three axis low-cost MEMS (Micro-Electro-Mechanical Systems) accelerometer. The first calibration algorithm uses traditional method (least square method). This method is furthermore compared with the second calibration method which uses differential evolution (DE) algorithm. The sensor error model (SEM) consists of three scale factors, three non-orthogonality errors, and three offsets. The performance of these methods is analysed in experiment on three axis low-cost accelerometer LSM303DLHC. The accelerometer readings were obtained in several precise angles. The experimental tests are conducted, and then the results are discussed and compared. The results show that the calibration error is least using DE algorithm.

Keywords: Accelerometer · Calibration · Differential evolution · Least square method · MEMS · Sensor

1 Introduction

Over the last decades, the advances in Micro-Electro-Mechanical System (MEMS) technologies have made a great role in many engineering applications. These advances enabled the usage of inertial sensor based on MEMS technologies in many areas such as information technology [1,2], car industry, military defence [3,4], and inertial localization and navigation systems [5,6].

These sensors such as accelerometers, gyroscopes, and magnetometers are relatively small, cheap, and have low power consumption. The accuracy of these sensors is influenced by a variety of errors (scale factor, bias, misalignment angles, temperature, etc.). Therefore, the proper calibration needs to be provided to minimize the errors. Calibration refers to procedures or techniques of measuring of known information and then estimating the corrective parameters that the measured output agree with the known value. The low-cost sensors are mostly factory trimmed which is insufficient.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_18

In this paper, we have focused only on the calibration of the low-cost three axis accelerometer. The accelerometers measures vibrations and acceleration of moving object.

Many researchers deals with the calibration of inertial accelerometers. Frosio et al. [7] presented on-field calibration for three axis accelerometer which not requires any special equipment. This procedure minimizes the bias, scale factor and cross axis errors. Woo et al. [8] used fuzzy inference system as a calibration procedure. The calibration method for three axis accelerometer and also magnetometer based on least square method were presented by Ammann et al. [9]. Olsson et al. [10] used sensor fusion with gyroscope. On measured data, maximum likelihood was then applied.

The aim of this paper is to evaluate the calibration methods which will account the systematic errors. The first method is least square method. This is the conventional method which is further described in [11]. The second method uses differential evolution (DE) algorithm for determination of transformation matrix and bias. The data for both methods were collected in several precise angles.

The reminder of the paper is organized as follows. In Sect. 2, the differential evolution algorithm is described. The hardware and software for data collection is briefly introduced in Sect. 3. The sensor error model and the accelerometer calibration is described in Sects. 4 and 5, respectively. The experimental results of accelerometer calibration is mentioned in Sect. 6.

2 Differential Evolution

Differential evolution is an optimization algorithm for heuristic search of function minimums introduced by R. Storn and K. Price in 1995 [12]. This optimization method is an evolutionary algorithm based on population, mutation and recombination. Differential evolution uses only four parameters which need to be set; therefore, it can be easily implemented. The parameters are Generations, NP, F and CR [13].

- **Generations** (Number of iterations) specifies the number of evolutionary cycles (generations) during which the entire population develops.
- **NP** (Number of population members) is a parameter which gives the size of the population. The value of this parameter cannot be lower than 4 because it is the minimum size at which the differential evolution algorithm still works. The optimal set up of this parameter is $5 \cdot D <= NP <= 10 \cdot D$ [12], where D is dimension of the problem.
- **CR** (Crossover probability). This parameter is a small value in range from 0 to 1. In case of separable function, this value is set close to 0 (clean copy of the fourth parent). Otherwise, it is set to the values close to 1 (random search).
- **F** (Mutation constant) is the last control parameter for differential evolution and its value ranges from 0 to 2.

The flow chart of differential evolution algorithm for the accelerometer calibration parameters estimation is shown in Fig. 1.

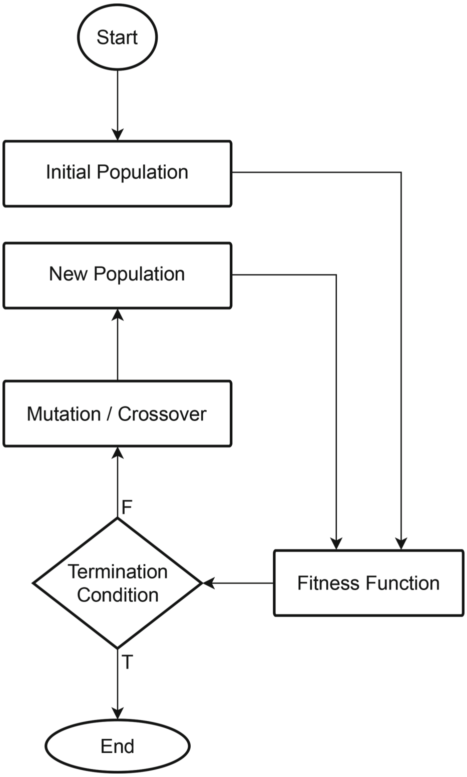


Fig. 1. Flow-chart of DE for parameters estimation.

3 Equipment

The experimental measurement chain (Fig. 2) includes control unit, inertial measurement unit (IMU) and software for data collection.

The control unit STEVAL-MKI109V2 is built up to provide platform for the evaluation of the MEMS modules. These modules can be connected via 24-pin expansion connector.

Furthermore, the unit consists high-performance 32-bit microcontroller STM32F103RET6, which is based on ARM technology, with 512 kB flash

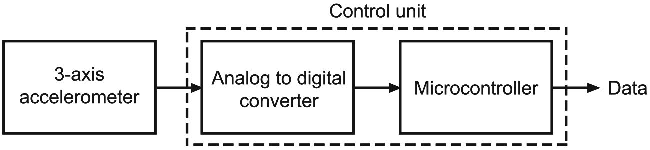


Fig. 2. Measurement chain.

memory functioning as a bridge between the MEMS modules and a graphical user interface (GUI) or dedicated software routines for customized applications.

To provide measurements, 10 axis inertial measurement unit STEVAL-MKI124V1 is connected to the control unit. The IMU includes three axis gyroscope with internal thermometer (L3GD20), three axis accelerometer and three axis magnetometer (LSM303DLHC), and barometer (LPS331AP). All these sensors are based on MEMS technology and they are factory tested, and trimmed. However, this factory calibration is appropriate only for basic applications. Advanced calibration had to be provided for application such as navigation systems.

Several different configurations allow for settings regarding specific usage. Sensor specifications are given in Table 1 and [14].

Table 1. Accelerometer Characteristics

	Accelerometer
Full scale	$\pm 2\,\text{g} - \pm 16\,\text{g}$
Sensitivity	1–12 mg/LSB
Min. zero level	±60 mg
Zero level vs. temp	$\pm 0.5\,\text{mg}/^\circ\text{C}$
Noise density	$220\,\mu\text{g}/\sqrt{Hz}$

To collect measured data, a PC is connected to the control unit using virtual serial port. On the PC, the drivers for interaction and configuration of sensors are installed. This software is called Unico STSW-MKI109W.

The collected data are processed in Wolfram Mathematica 10 and then the differential evolution algorithm, programmed in Lua language, is applied.

4 Sensor Error Model

The three axis accelerometer readings are influenced by many sources of error like stochastic biases, installation errors, and wide-band measurement noise. These errors in accelerations lead to continual grow of the error in velocity and distance, due to integration.

The mathematical model, which describes the accelerometer readings, is expressed by

$$A = M_m \cdot S \cdot (M + O + n) \tag{1}$$

In this model, the variables M_m, and S are matrices which interpret misalignment errors, and scale factors, respectively. O and n are vectors representing biases and wide-band noise which influences the acceleration vector M in the local reference system (Fig. 3).

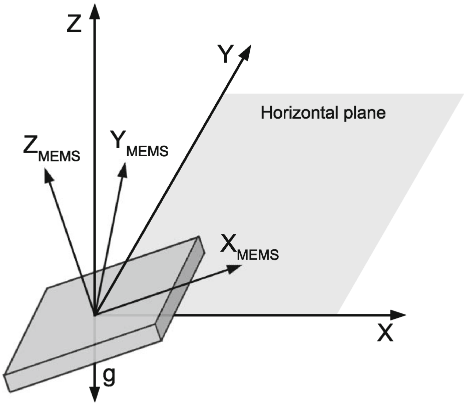


Fig. 3. Accelerometer local reference system.

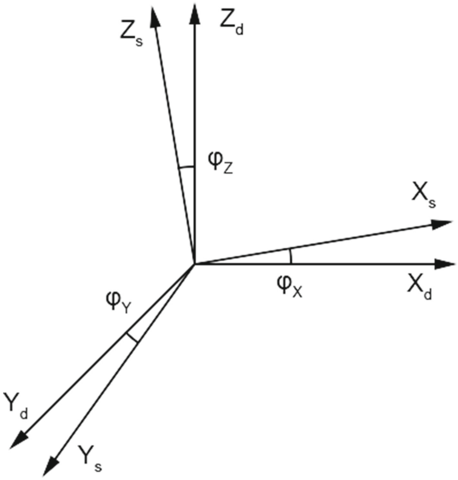


Fig. 4. Misalignment error

1. **Misalignment error** is defined as angles between the accelerometer axis X_s, Y_s, Z_s and the device body axis X_d, Y_d, Z_d. This caused by imperfect mounting of sensor on the PCB (printed circuit board) (Fig. 4).

$$M_m = \begin{bmatrix} 1 & m_{xy} & m_{xz} \\ m_{yx} & 1 & m_{yz} \\ m_{zx} & m_{zy} & 1 \end{bmatrix} \tag{2}$$

2. **Scale factor error** corresponds to constants of proportional relationship between the input and output of the accelerometer (Fig. 5). The scale factor can be modelled as

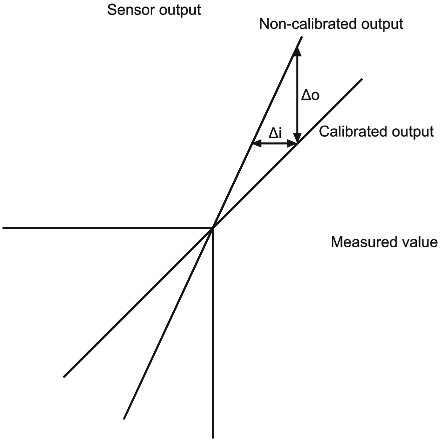


Fig. 5. Scale factor error

$$S = diag\left(s_x\ s_y\ s_z\right) \tag{3}$$

3. **Bias** is a constant offset in the raw accelerometer readings that changes randomly after turning the device on.

$$O = \left(o_x\ o_y\ o_z\right)^T \tag{4}$$

5 Accelerometer Calibration

The calibration requires measurements in several different static positions to determine scale factor, misalignment errors and bias. That means 9 unknown calibration parameters - three misalignment angles, three scale factors, and three biases. The measured datasets consists of 10 different positions for each sensitive axis (see Fig. 6).

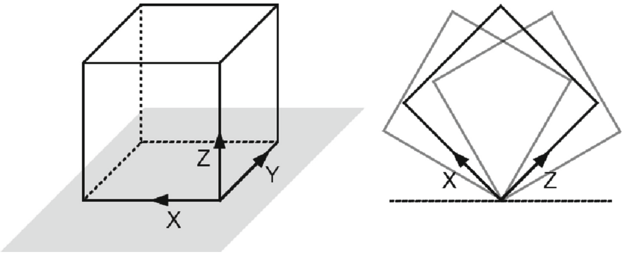


Fig. 6. Static positions for accelerometer calibration

The average of squared errors have been used as the fitness function where the error is the difference between the calculated output from current parameters and the true value. Therefore, the fitness function of the DE algorithm is modelled as

$$F = \sum_{i=1}^{n} \left(\sqrt{(X)^2 + (Y)^2 + (Z)^2} - 1 \right)^2 \tag{5}$$

where X, Y, and Z are calibrated values and value 1 is the gravitation field given in g-force acceleration ($1g = 9.81m \cdot s^{-2}$).

The calibrated values account for bias offset (O_X, O_Y, O_Z), scale factor (S_X, S_Y, S_Z), and misalignment error (α, β, γ). The equations for calculations of such errors are showed in (6), (7) and (8)

$$X = (R_X - O_X) \cdot S_X \tag{6}$$

$$Y = (R_X - O_X) \cdot \alpha + (R_Y - O_Y) \cdot S_Y \tag{7}$$

$$Z = (R_X - O_X) \cdot \beta + (R_Y - O_Y) \cdot \gamma + (R_Z - O_Z) \cdot S_Z \tag{8}$$

The model for calculation of misalignment errors is depicted in Fig. 7.

Table 2 shows the set-up of differential evolution. The parameter NP is set to the upper limitation suggested by [12] to $10 \cdot D$, where the dimension of the problem is 9 (the number of unknown calibration parameters). The number of generations is given by the experiment that showed the optimal value 300 (slow convergence rate to minimum). The parameters F and CR was also given by an experiment. The best set-up of DE is the subject of further research. We use version of DE known as DERand1Bin.

The t-test was conducted to provide statistical evidence for the calibration method with DE and least square method.

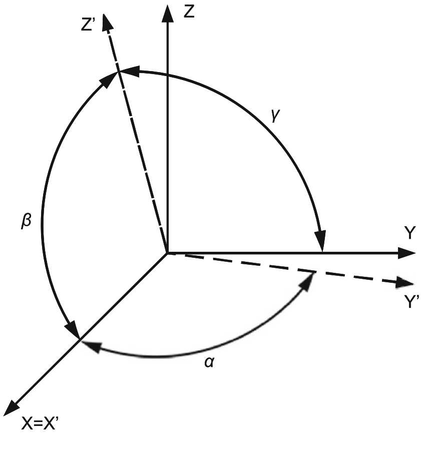


Fig. 7. Model for misalignment error.

Table 2. Set-up of differential evolution.

Parameter	Value
NP	90
Generations	300
F	0.5
CR	0.9

```
Welch's t-test
data: Least square method and DE
t = −2.8925, df = 46.079, p-value = 0.005817
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
−0.010040984      −0.001800708
sample estimates:
mean of x mean of y
0.9935837         0.9995045
```

The p-value is lower than 0.05; therefore, the null hypothesis can be rejected. The alternative hypothesis is accepted; thus, the DE could have lower error than the least square method.

6 Experimental Results

The proposed calibration procedure for low-cost three axis accelerometer has been applied on accelerometer module LSM303DLHC. The parameters of this accelerometer are shown in Table 1. The output readings have been collected by the computer at a rate of 200 MHz which is more than enough for human motion capture. The data has been measured for at least 10 s in each position and then filtered by low pass filter to remove high frequency noise.

To evaluate the performance of the calibration method using DE algorithm, we used the root square mean error (RMSE) even for least square method and raw data. The smaller is value of RMSE, the better is performance of calibration method.

$$RMSE = \sqrt{\frac{1}{N} \cdot \sum_{i=1}^{N} (x_i - \hat{x_i})^2}, \tag{9}$$

where N is equal to the number of samples, $\hat{x_i}$ is measured gravity, and x_i is true magnitude of gravitational field which is $1g$.

The scalar error is listed in Table 3. The RMSE is the least with DE and it is two times better than least square method.

Table 3. RMSE using least square method and DE.

Before calibration	Least square method	DE
0.0566	0.0122	0.005

7 Conclusion

In this paper, we presented a calibration method using differential evolution algorithm. This calibration method determines parameters of scale factor, misalignment angles, and bias. Then, it has been tested on module LSM303DLHC which consists of three axis accelerometer. The experimental results show, that the errors can be minimized only to the random errors. The comparison shows that the DE algorithm has lower error.

In future work, the best set-up of parameters of DE algorithm will be tested.

Acknowledgments. This work was supported by Internal Grant Agency of Tomas Bata University in Zlin under the project No. IGA/FAI/2017/007.

References

1. Oboe, R.: Use of MEMS based accelerometers in hard disk drives. In: Proceedings of the 2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (Cat. No.01TH8556), vol. 2, pp. 1142–1147. IEEE (2001). http://ieeexplore.ieee.org/document/936863/
2. Abramovitch, D.Y., Hsu, G.: Mitigating rotational disturbances on a disk drive with mismatched linear accelerometers. In: 2015 IEEE Conference on Control Applications (CCA), pp. 1473–1478. IEEE, September 2015. http://ieeexplore.ieee.org/document/7320819/
3. Nelson, G., Rajamani, R.: Accelerometer based acoustic control: enabling auscultation on a black hawk helicopter. IEEE/ASME Trans. Mechatron., 1 (2016). http://ieeexplore.ieee.org/document/7553467/
4. Wei, X.: Autonomous control system for the quadrotor unmanned aerial vehicle. In: 2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), pp. 796–799. IEEE, August 2016. http://ieeexplore.ieee.org/document/7733984/
5. Wu, C., Mu, Q., Zhang, Z., Jin, Y., Wang, Z., Shi, G.: Indoor positioning system based on inertial MEMS sensors: design and realization. In: 2016 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), pp. 370–375. IEEE, June 2016. http://ieeexplore.ieee.org/document/7574852/
6. Zhang, X., Zhang, R., Guo, M., Cheng, G., Niu, S., Li, J.: The performance impact evaluation on bias of gyro and accelerometer for foot-mounted INS. In: 2015 12th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), pp. 1541–1546. IEEE (2015). http://ieeexplore.ieee.org/document/7494468/

7. Frosio, I., Pedersini, F., Borghese, N.: Autocalibration of MEMS Accelerometers. IEEE Trans. Instrum. Meas. **58**(6), 2034–2041 (2009). http://ieeexplore.ieee.org/document/4655611/
8. Woo, S., Kim, J., Kim, J., Kim, S.: Calibration of accelerometer using fuzzy inference system. In: 11th International Conference on Control, Automation and Systems (ICCAS), pp. 1448–1450 (2011)
9. Ammann, N., Derksen, A., Heck, C.: A novel magnetometer-accelerometer calibration based on a least squares approach. In: 2015 International Conference on Unmanned Aircraft Systems (ICUAS), pp. 577–585. IEEE, June 2015. http://ieeexplore.ieee.org/document/7152338/
10. Olsson, F., Kok, M., Halvorsen, K., Schon, T.B.: Accelerometer calibration using sensor fusion with a gyroscope. In: 2016 IEEE Statistical Signal Processing Workshop (SSP), pp. 1–5. IEEE, June 2016. http://ieeexplore.ieee.org/document/7551836/
11. STMicroelectronics, Application note: using LSM303DLH for a tilt compensated electronic compass, STMicroelectronics, Technical report (2010)
12. Storn, R., Price, K.: Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical report (1995)
13. Storn, R.: On the usage of differential evolution for function optimization. In: Proceedings of North American Fuzzy Information Processing, pp. 519–523. IEEE (1996). http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=534789
14. STMicroelectronics, Data brief: STEVAL-MKI124V1, p. 4 (2013). http://www.st.com/web/en/resource/technical/document/data_brief/DM00052740.pdf

Hierarchical Rough Fuzzy Self Organizing Map for Weblog Prediction

Arindam Chaudhuri[1(✉)] and Soumya K. Ghosh[2]

[1] Samsung R&D Institute Delhi, Noida 201304, India
arindam_chau@yahoo.co.in
[2] Department of Computer Science Engineering,
Indian Institute of Technology Kharagpur, Kharagpur 721302, India
skg@iitkgp.ac.in

Abstract. Web servers play a vital role in conveying knowledge and information to end users. With rapid growth of WWW over past decades discovering hidden information about the usage pattern is critical towards determining effective strategies as well as to optimize server usage. Most of the available server analysis tools provide statistical data only without much useful information. Mining useful information becomes challenging task when user traffic data is huge and keeps on growing. In this work we propose hierarchical rough fuzzy self–organizing map (HRFSOM) to analyze useful information from the statistical data through weblog analyzer. We use cluster information generated by HRFSOM for data analysis and a variant of takagi sugeno fuzzy inference system (TSFIS) to predict daily and hourly traffic jam volumes. The experiments are performed using web user access sample patterns available at Yandex Personalized Web Search Challenge where statistical weblog data is generated by AWStats web access log file analyzer. The proposed classifier has superior clustering accuracy compared to other classifiers. The experimental results demonstrate the efficiency of proposed approach.

Keywords: Weblog prediction · Weblog data · SOM · RFSOM · HRFSOM

1 Introduction

In the present competitive business scenario web has become a place where people of all ages, languages and cultures conduct their digital lives [1]. The web usage entails a wide array of devices, networks and applications [2, 3]. When searching, creating and disseminating information, users leave behind great deal of data revealing their information needs, attitudes and personal facts. The web designers collect these artifacts in weblogs for subsequent analysis. WWW is always expanding with rapid increase of information transaction from web users. For web administrators, discovering hidden information about users' access patterns improves web information service performance quality. From business point of view, knowledge obtained from access patterns manages e-business services. The statistical data obtained from weblog files provide information explicitly. The analysis relies on past usage patterns, shared content degrees and inter-memory associative links. This leads to content intelligence

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_19

improving overall system quality. The pattern discovery of web usage mining consists of statistical analysis, clustering etc. Most of the existing research focuses on finding patterns with insignificant pattern analysis. Some of the important weblog analysis techniques are conceptual framework, phenomenology, content analysis, discourse analysis etc.

Considering different weblog analysis techniques access patterns available at Yandex Personalized Web Search Challenge [4] are used to predict daily and hourly traffic jams. The predicted results provide information for decision making activities. To achieve this hierarchical rough fuzzy self-organizing map (HRFSOM) is proposed to cluster and discover patterns from data that is used for statistical analysis. To make analysis more intelligent clustered data is used for prediction. A variant of takagi sugeno fuzzy inference system (TSFIS) [2] explores prediction of average daily and hourly traffic jams. The statistical data from sample patterns is provided by AWStats log file analyzer [5]. The generated data covers different aspects of users' access log records, weekly based reports, domain summary navigation summary etc. The challenge lies in finding relevant hidden information through extraction of patterns. The information analysis and prediction from huge datasets entails requirement of hybrid intelligent systems. The major contributions of this work includes: (a) input representation of SOM [6] as rough granules or nuggets (b) rough fuzzy sets [7] formulation to extract domain knowledge from data (c) training HRFSOM to cluster data and (d) using variant of TSFIS to predict daily and hourly traffic jam. This paper is organized as follows. In Sect. 2 computational of HRFSOM is highlighted. This is followed by experiments and results in Sect. 3. Finally in Sect. 4 conclusions are given.

2 Computational Method

In this section mathematical framework of proposed HRFSOM model [2] is presented. The schematic representation of prediction system is given in Fig. 1.

2.1 Problem Description

The research problem entails in predicting the daily and hourly traffic jams on ever growing traffic data. In order to achieve this prediction task, we propose a SOM based predictor viz RFSOM to analyze the web user access sample patterns at Yandex Personalized Web Search Challenge [4]. The statistical data from sample patterns is provided by AWStats web access log file analyzer [5]. The clusters generated by RFSOM are used by takagi sugeno fuzzy inference system (TSFIS) variant for prediction. The prediction task performed on daily and hourly traffic jams provide information for several decision making activities.

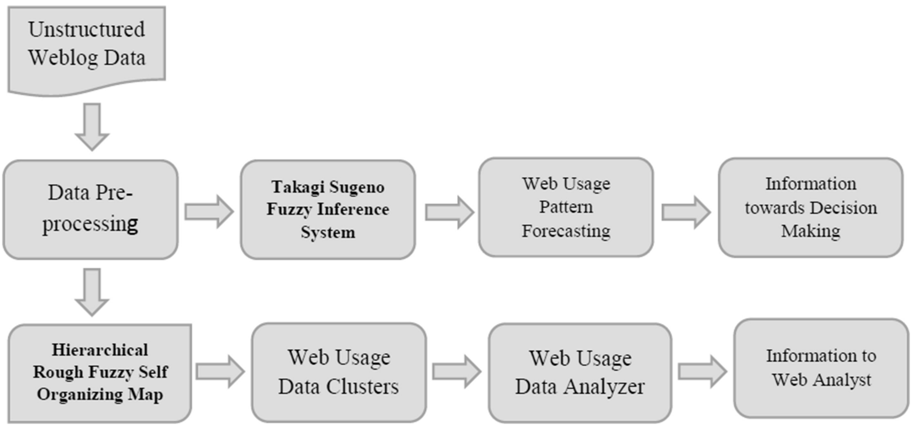

Unstructured Weblog Data
Data Pre-processing
Takagi Sugeno Fuzzy Inference System
Web Usage Pattern Forecasting
Information towards Decision Making
Hierarchical Rough Fuzzy Self Organizing Map
Web Usage Data Clusters
Web Usage Data Analyzer
Information to Web Analyst

Fig. 1. The schematic representation of prediction system for web pattern analysis

2.2 Datasets

The experimental data is taken from Yandex Personalized Web Search Challenge 2014 which re-ranked web documents using personal preferences [4]. Here an opportunity was provided to consolidate and scrutinize the work from industrial labs on personalizing web search using user-logged search behavior context. It provided fully anonymized dataset shared by Yandex which had anonymized user ids, queries, query terms, urls and clicks. Each user record is specified in terms of user identification, query and url. The content tag contains an individual record. The record is followed by query term and clicks. Each of these records in dataset has recorded entity and level of prediction involved. After performing experiments with dataset more than 70000 records are labelled manually. The total dataset log contained 167,413,039 labelled records. The data is pre-processed to fit record mentioned in specific buckets. The logs are about 2 years old. The queries and users are sampled from only one region. The sessions containing queries with unwanted intent detected with Yandex classifier are removed considering the top-k most popular queries where k is user defined.

2.3 Rough Fuzzy Self-organizing Map for Weblog Mining

In this section RFSOM [2, 8] is evaluated in terms of rough fuzzy sets and SOM. It performs clustering and discovers patterns from data which are used for statistical analysis. The clustered data is used to predict daily and hourly traffic using TSFIS variant. The experimental data is adapted from single site weblog data generated by AWStats web access log file analyzer [5]. The web user access data is taken from 16th May 2010 to 16th May 2011. This raw data is unstructured in nature and is converted to structured format. After initial analysis statistical data comprising of number of domain requests, daily requests and hourly page traffic volume is selected to develop cluster models for finding web users usage patterns. The data is cleaned by removing irrelevant noise. The datasets are scaled to two tone format [2]. Other inputs such as number of users and total processing time are considered to distinguish temporal sequence of data. The data having most recent access are given higher index. The preprocessed data is presented to RFSOM for clustering and discovering patterns. The major phases are:

Representation of input vectors of SOM in terms of rough granules or nuggets: SOM input vector is described in terms of rough granules lower, median and upper. The human mind performs tasks based on perceptions presented through rough granules or nuggets [2]. A rough granule is group of patterns defined by generalized constraint $Y\, rs\, R$; R is constrained relation, rs is random set constraint combining probabilistic and possibilistic constraints and Y is rough set valued random variable. A pattern $y \in U$ is assigned value with membership function $\mu_Y^A(y)$ as:

$$\mu_Y^A(y) = \frac{\left\| [y]_A \cap Y \right\|}{\left\| [y]_A \right\|} \quad \text{for}\, y \in U \tag{1}$$

In Eq. (1) membership values are defined by π-membership function as:

$$\pi(y, C, \lambda) = \begin{cases} \quad 2\left(1 - \frac{\|y-C\|_2}{\lambda}\right)^2 & \text{for}\ \frac{\lambda}{2} \le \|y - C\|_2 \le \lambda \\ \quad 1 - 2\left(\frac{\|y-C\|_2}{\lambda}\right)^2 & \text{for}\, 0 \le \|y - C\|_2 \le \frac{\lambda}{2} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

In Eq. (2) $\lambda > 0$ is scaling factor of π function with C as central point and $\|\cdot\|_2$ is Euclidian norm. The π functions parameters are governed by p patterns set with n features $\left\{P_{ij}; i = 1, \ldots\ldots, p, j = 1, \ldots\ldots, n\right\}$. Here P_{jmin_m} and P_{jmax_m} are minimum and maximum values respectively along j^{th} feature for p patterns. The outliers affect parameters, centre and scaling factor of π function. Their effect is reduced by taking average of feature values of p patterns along j^{th} feature P_j which is centre of linguistic term c_{m_j}. The average pattern values having labels in ranges $\left[P_{j_{min_m}}, c_{m_j}\right)$ and $\left(c_{m_j}, P_{j_{max_m}}\right]$ are defined as middle points of linguistic terms *lower* and *upper* by c_{low_j} and c_{up_j} respectively. Similarly patterns with values in above ranges can be considered along j^{th} axis $P_{j_{min_{low}}} = P_{j_{min_m}}$, $P_{j_{max_{low}}} = c_{m_j}$, $P_{j_{min_{up}}} = c_{m_j}$ and $P_{j_{max_{up}}} = P_{j_{max_m}}$ [2]. To incorporate granular concept a n-dimensional pattern is represented as 3n-dimensional linguistic vector. If $Ft_{i1}, \ldots\ldots, Ft_{in}$ represent n features of i^{th} pattern $\boldsymbol{Ft_i}$ with μ being π-membership function then rough granule of features is:

$$\begin{aligned} \boldsymbol{Ft_i} = [&\left\{\mu_{lower(Ft_{i1})}(\boldsymbol{Ft_i}), \mu_{median(Ft_{i1})}(\boldsymbol{Ft_i}), \mu_{upper(Ft_{i1})}(\boldsymbol{Ft_i})\right\}, \ldots\ldots, \\ &\left\{\mu_{lower(Ft_{in})}(\boldsymbol{Ft_i}), \mu_{median(Ft_{in})}(\boldsymbol{Ft_i}), \mu_{upper(Ft_{in})}(\boldsymbol{Ft_i})\right\}] \end{aligned} \tag{3}$$

Crystallization of linguistic input data based on α-cut: Once input vectors of SOM are represented as rough granules, linguistic input is crystallized. This is done in two phases through computation of similarity matrix and generation of crystallization structures. The algorithm for first phase defines pairwise similarity matrix along linguistic inputs through rough set connectives. The similarity matrix develops crystallization structures based on α-value $(0<\alpha<1)$. In second phase crystallization structures are determined in q groups. Both algorithms are available in [2]. The resultant structures (q groups) are partitions or clusters. These partitions are arranged in decreasing order according to group size. The top m user defined groups based on their size are selected for every α-value in all q groups. The compactness of first m groups for every α-value are calculated using rough fuzzy entropy [7] and crystallization structures. For any particular α-value lowest average rough entropy are accepted.

Extraction of domain knowledge through rough fuzzy sets: The lowest average rough fuzzy entropy values [7] are presented to decision system DT to extract domain knowledge. Let p be the number of patterns in all m groups obtained using selected α-value. These p patterns from m groups $\{y_1, \ldots., y_p\}$ are presented to decision system $DT = (U, B \cup \{v\})$ where U and B represent universe and attributes $\{b_1, \ldots., b_{3n}\}$ respectively. Here each attribute is constructed by considering corresponding decision from $3n$-dimensional linguistic vectors from Eq. (3). The decision attribute d defined as Y_j; $j = 1, \ldots., m$ corresponding to pattern is assigned according to its group. Each Y_j is treated as decision class. Each pattern $y_i \in U$ is classified by its decision class. The rough fuzzy reflexive relation R_b between any two patterns p and q in U with respect to quantitative attribute $b \in B$ is:

$$R_b = \begin{cases} max\left(min\left(\frac{b(q)-b(p)+2\sigma_{b_{j_1}}}{2\sigma_{b_{j_1}}}, \frac{b(p)-b(q)+2\sigma_{b_{j_1}}}{2\sigma_{b_{j_1}}}\right), 0\right) & \text{if } b(p),\, b(q) \in R_d\left(Y_{j_1}\right) \\ max\left(min\left(\frac{b(q)-b(p)+2\sigma_{b_{j_2}}}{2\sigma_{b_{j_2}}}, \frac{b(p)-b(q)+2\sigma_{b_{j_2}}}{2\sigma_{b_{j_2}}}\right), 0\right) & \text{if } b(p) \in R_d\left(Y_{j_1}\right),\, b(q) \in R_d\left(Y_{j_2}\right) \wedge j_1 \neq j_2 \end{cases} \tag{4}$$

In Eq. (4), $j_1 = 1, \ldots., m$, $j_2 = 1, \ldots., m$ and $\sigma_{b_{j_1}}$, $\sigma_{b_{j_2}}$ represent standard deviation of decision classes Y_{j_1}, Y_{j_2} respectively; $b(p), b(q) \in R_d\left(Y_{j_1}\right)$ which implies patterns p, $q \in Y_{j_1}$ with respect to decision attribute $\{d\}$ where $b \in \{d\}$. Again $b(p) \in R_d\left(Y_{j_1}\right)$ and $b(q) \in R_d\left(Y_{j_2}\right)$ imply that patterns p, q belong to two different decision classes Y_{j_1}, Y_{j_2} respectively. The decision system DT contains p decision classes. The $3n$-dimensional vectors M_{kj} and S_{kj}; $j = 1, \ldots., 3n$ are mean and standard deviation respectively of patterns belonging to k^{th} decision class. The weighted distance of pattern $\boldsymbol{Ft_i}$; $i = 1, \ldots., p$ from k^{th} decision class is:

$$Z_{ik} = \sqrt{\sum\nolimits_{j=1}^{n} \left[\frac{Ft_{ij} - M_{kj}}{S_{kj}}\right]^2} \forall\, k = 1, 2, 3, \ldots\ldots., p \tag{5}$$

In Eq. (5), Ft_{ij} is j^{th} part of i^{th} pattern and the pattern membership is:

$$\mu_k(\boldsymbol{Ft_i}) = \frac{1}{1 + \left(\frac{Z_{ik}}{fz_d}\right)^{fz_e}} \tag{6}$$

In Eq. (6), fz_d and fz_e are rough entities. When a pattern has different membership values then its decision attribute becomes quantitative. This is shown in two ways. The membership values of all patterns in k^{th} class to its own class is $E_{kk} = \mu_k(\boldsymbol{Ft_i})$ if $k = v$ and membership values of all patterns in k^{th} class to other classes is $E_{kv} = 1$ if $k \neq v$; k, $v = 1, \ldots., p$. For any $C \subseteq B$ rough positive region is defined based on C-indiscernibility relation R_C for $p \in U$ as:

$$POS_c(q) = \left(\cup_{p \in U} R_C \downarrow R_d p \right)(q) \, \forall q \in U \tag{7}$$

Incorporating domain knowledge in SOM: The decision table *DT* explains granulation concept by partition and rough fuzzy set approximations based on rough reflexive relation. By this knowledge data is extracted and incorporated into SOM which is used for competitive learning. The knowledge about encoding procedure considers decision table *DT* with its set of conditional attributes, decision attributes, set of patterns and labeled values of patterns corresponding to $3n$-dimensional conditional attributes. The decision table *DT* extracts domain knowledge about data using following steps: (a) Generate rough reflexive relational matrix on all possible patterns pairs and obtain additional granulation structures (b) Using rough reflexive relational matrix compute memberships belonging to lower approximation of every pattern for each conditional attribute (c) Calculate rough positive region of every pattern for each conditional attribute (d) Calculate degree of dependency of each conditional attribute with respect to each decision class and assign resulting dependency factors as initial weights between input layer and (user defined clusters) output layer nodes.

Training RFSOM and clustering the data: After RFSOM is developed its training is done through following steps: (a) Transform input data into 3-dimensional granular space (b) Choose initial RFSOM connection weights using rough fuzzy sets (c) Train RFSOM through competitive learning (d) Partition data into clusters of granulation structures (e) Update weights connecting input layer to winning node and neighboring nodes (f) Repeat steps (c)–(e) by adjusting connection weights and neighborhood size (g) Map each term to node and label winning nodes to make output space ordered. To select RFSOM parameters trial and error approach is adopted. This reduces normalized distortion and quantization error. The clustering results accuracy obtained from RFSOM are better than SOM and fuzzy SOM (FSOM) algorithms [2, 6, 8].

2.4 Hierarchical Bidirectional Recurrent Neural Network for Semantic Analysis

The hierarchical version of RFSOM viz HRFSOM [2] is proposed here. The computational benefits [2] serve the major motivation. HRFSOM is different from RFSOM in terms of efficient classification accuracy based on similarities and running time when volume of data grows [2]. The model architecture is shown in Fig. 2. HRFSOM architecture correlates data behavior across multiple features of relevance. This facilitates distribution of computational overheads in predictor construction. At 1st and 2nd layers relatively small RFSOMs are utilized (6 × 6). The number of RFSOMs are increased at layers 3, 4 and 5. RFSOMs in last layer are constructed over subset of examples for which neuron in 5th layer is Best Matching Unit (BMU). RFSOMs at last layer can be larger (40 × 40) than used in 1st to 5th layers. It improves resolution and discriminatory capacity of RFSOM with less training overhead. Building HRFSOM requires several data normalization operations. This provides for initial temporal pre-processing and inter-layer quantization between 1st to 5th layers. The pre-processing provides suitable representation for data and supports time based representation. The 1st SOM layer treats each feature independently with each data instance mapped to sequential values. In case of temporal representation standard RFSOM has no capacity to recall histories of patterns directly. A shift register of length l is employed in which tap is taken at predetermined repeating interval k such that $l \% k = 0$ where % is modulus operator. The 1st level RFSOMs only receive values from shift register. Thus, as each new connection is encountered (at left), content of each shift register location is transferred one location (to right) with previous item in l^{th} location being lost. In case of n-feature architecture it is necessary to quantize number of neurons between 1st to 5th level RFSOMs. The purpose of 2nd to 5th level RFSOM is to provide an integrated view of input feature specific RFSOMs developed in 1st layer. There is potential for each neuron in 2nd to 5th layer RFSOM to have an input dimension defined by total neuron count across all 1st layer RFSOMs. This is brute force solution that does not scale computationally. Given topological ordering provided by RFSOM, neighboring neurons respond to similar stimuli. The topology of each 1st layer SOM is quantized in terms of fixed number of neurons using potential function clustering algorithm [2, 6, 8]. This reduces number of inputs in 2nd to 5th layers of RFSOM. The neurons in 4th layer acts as BMU for examples with same class label thus maximizing detection rate and minimizing false positives. However, there is no guarantee for this. In order to resolve this 4th layer SOM neurons that act as BMU for examples from more than one class are used to partition data. The 5th layer RFSOMs are trained on subsets of original training data. This enables size of 5th layer RFSOMs to increase which improves class specificity while presenting reasonable computational cost. Once training is complete 4th layer BMUs acts to identify which examples are forwarded to corresponding 5th layer RFSOMs on test dataset. A decision rule is required to determine under what conditions classification performance of BMU at 4th layer RFSOM is judged sufficiently poor for association with 5th layer RFSOM. There are several aspects that require attention such as minimum acceptable misclassification rate of 4th layer BMU relative to number of examples labeled at 4th layer BMU and number of examples 4th layer BMU represent. The basic implication is that there must be optimal number of connections associated with 4th layer BMU for training of corresponding 5th layer RFSOM and misclassification rate over examples associated with 4th layer BMU exceeds threshold. HRFSOM is characterized in terms of success probability in recovering true hierarchy H^* and runtime complexity. Some restrictions are placed to similarity function S [2] such that similarities scale with hierarchy upto some random noise: (a) For each $y_i \in Cs_j \in Cs^*$ and $j' \neq j$: $\min_{y_p \in Cs_j} \mathbb{Exp}[S(y_i, y_p)] - \max_{y_p \in Cs'_j} \mathbb{Exp}[S(y_i, y_p)] \geq \gamma > 0$. Here expectations are taken with respect to noise on S. (b) S2 For each $y_i \in Ct_j$, a set of V_j words of size v_j drawn uniformly from Cs_j satisfies:

$\mathbb{Prob}\left(\min_{y_p \in Cs_j} \mathbb{Exp}[S(y_i, y_p)] - \sum_{y_p \in V_j} \frac{S(y_i, y_p)}{v \cdot} > \epsilon\right) \leq 2e^{\left\{\frac{-2v_j\epsilon^2}{\sigma^2}\right\}}$. Here $\sigma^2 \geq 0$ parameterizes noise on similarity function S. From viewpoint of feature learning stacked RFSOMs extracts temporal features of sequences in weblog data. Various trade-offs are done towards improving representation ability and avoiding data over fitting. It is easy to overfit the network with limited data training sequences. This algorithm can be fine-tuned with heuristics.

3 Experiments and Results

In this section experimental results are presented for weblog prediction. When data is clustered by HRFSOM it is taken up by WUDA to discover request patterns and daily requests clusters. The fuzzy inference system uses patterns to infer meaningful information from data. During training process HRFSOM generated 5 clusters for hourly number of requests as shown in Table 1. The clusters generated in Table 1 are scattered in nature with respect to hourly requests and are almost identical. This classification ambiguity is resolved through WUDA by hourly page traffic volume and page requests. The clusters 4 and 5 have higher request and pages than clusters 1, 2 and 3. Using conventional web log analyzers it is difficult to understand daily traffic pattern. As a result of this data is clustered based on total activity for each day of week using volume of daily requests, pages and index value as input features. The training through HRFSOM generated 7 clusters as shown in Table 2. The clusters are separated according to access time as revealed by WUDA. After patterns are discovered through

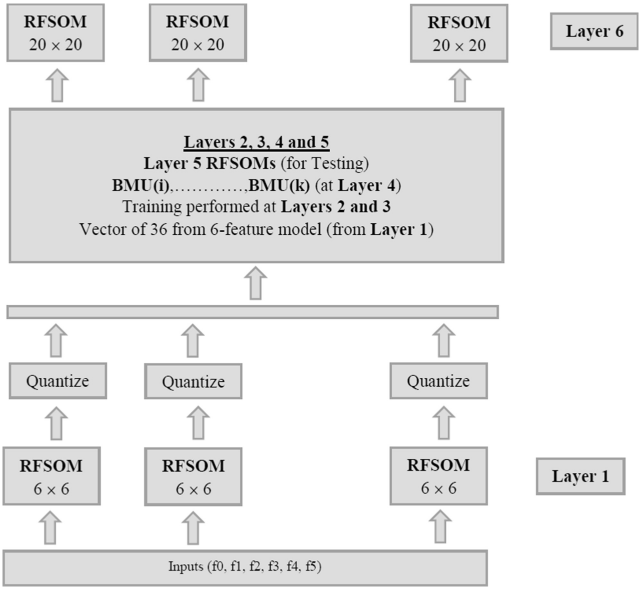

RFSOM
20 × 20
RFSOM
20 × 20
RFSOM
20 × 20
Layer 6
Layers 2, 3, 4 and 5
Layer 5 RFSOMs (for Testing)
BMU(i).............BMU(k) (at Layer 4)
Training performed at Layers 2 and 3
Vector of 36 from 6-feature model (from Layer 1)
Quantize
Quantize
Quantize
RFSOM
6 × 6
RFSOM
6 × 6
RFSOM
6 × 6
Layer 1
Inputs (f0, f1, f2, f3, f4, f5)

Fig. 2. The architecture of proposed HRFSOM model

Table 1. Hourly request clusters using HRFSOM

Clusters	Contribution percentage	Cluster contents (Percentage)
Cluster 1	18	14, 15, 16, 17, 18, 19, 20
Cluster 2	16	1, 2, 3, 4, 5, 6, 7, 21, 22, 23
Cluster 3	14	7, 8, 9, 10, 11, 13, 14, 20, 21, 22, 23
Cluster 4	25	9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21
Cluster 5	27	0, 1, 2, 3, 4, 5, 6, 7, 8, 10, 18, 19, 20, 21, 22, 23

Table 2. Clustering of days using HRFSOM

Clusters	Contribution percentage	Cluster contents (Percentage)
Cluster 1	18	Sunday, Monday, Tuesday, Wednesday, Thursday
Cluster 2	14	Thursday, Friday
Cluster 3	30	Sunday, Monday, Tuesday, Wednesday, Thursday
Cluster 4	5	Monday, Tuesday, Wednesday
Cluster 5	24	Thursday, Friday
Cluster 6	5	Saturday, Sunday
Cluster 7	4	Saturday, Sunday

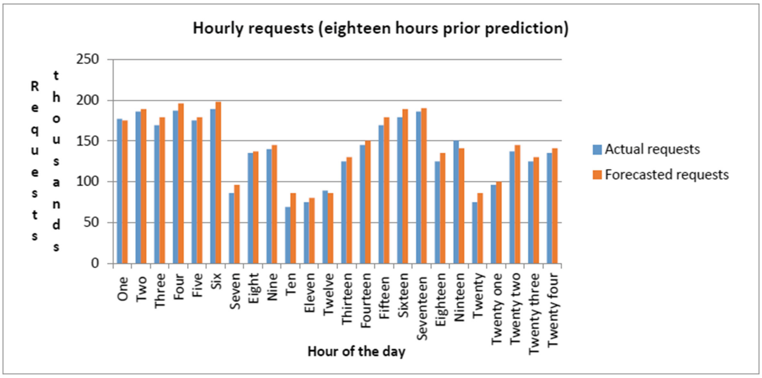
WUDA prediction is performed using variant of TSFIS [2]. The computing framework is based on rough fuzzy sets, fuzzy if then rules and fuzzy reasoning. The fuzzy rule is constituted by weighted linear combination of crisp inputs [2]. TSFIS performs interpretation in form of simple if then rules. Adaptive neuro fuzzy inference system (ANFIS) [8] and least mean squares (LMS) estimation [8] are used to fine tune antecedent and consequent rule parameters of TSFIS respectively. The data from 16th May 2010 to 31st December 2010 and data from 1st January 2011 to 16th May 2011 are used for training and testing purposes respectively. Grid partitioning is used to generate

Table 3(a). Prediction of daily traffic jam

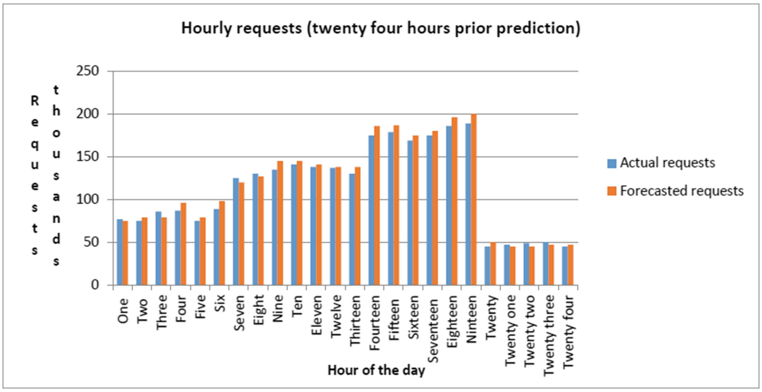
Prediction period (days)	Root Mean Squared Error (RMSE)			
	Takagi Sugeno Fuzzy Inference System			
	With cluster location information		Without cluster location information	
	Training	Testing	Training	Testing
One	0.01569	0.03096	0.06475	0.09550
Two	0.05365	0.06998	0.10269	0.12747
Three	0.05067	0.05995	0.12845	0.14259
Four	0.05541	0.06886	0.11669	0.12998
Five	0.06847	0.07886	0.12486	0.14447
Six	0.06745	0.07465	0.12325	0.14396

Table 3(b). Prediction of hourly traffic jam

Prediction period (hours)	Root Mean Squared Error (RMSE)			
	Takagi Sugeno Fuzzy Inference System			
	With cluster location information		Without cluster location information	
	Training	Testing	Training	Testing
One	0.03434	0.03537	0.09579	0.09986
Six	0.04559	0.05477	0.09789	0.98989
Twelve	0.06519	0.06769	0.10095	0.11517
Eighteen	0.05596	0.06777	0.10477	0.10986
Twenty Four	0.05736	0.06696	0.10595	0.10769



(a)



(b)

Fig. 3. **(a)** Prediction of traffic jam eighteen hours ahead **(b)** Prediction of traffic jam twenty four hours ahead

Table 4. The comparison of clustering accuracy of SOM, FSOM, RFSOM and HRFSOM

Analysis algorithms	Accuracy percentage (with respect to web user access patterns)
SOM	70
FSOM	79
RFSOM	86
HRFSOM	96

initial rule base. Only small number of membership functions are required for each input and 2D spaces are partitioned using trapezoidal membership functions [2]. Alongwith regular inputs, cluster location information from HRFSOM output is also used. Based on these inputs fuzzy inference models are developed to predict web traffic volume on hourly and daily basis. Once traffic volume of a particular day is available the model predicts daily traffic upto 7 days ahead. The Tables 3(a) and (b) summarizes performance of fuzzy inference system for training and testing data in days and hours respectively. The prediction of hourly traffic is done upto 24 h ahead. The Fig. 3(a) and (b) represent test results for 18 and 24 h ahead prediction of hourly web traffic volume. The accuracy of clustering results of web user access patterns obtained from HRFSOM are highly significant are highly significant as shown in Table 4 when compared to other clustering algorithms such as SOM, fuzzy self-organizing map (FSOM) [2] and RFSOM.

4 Conclusion

The knowledge discovery from data in terms of relevant user information and access patterns allows organizations to predict user's future access patterns. This helps in further development, planning and maintenance towards advertising campaigns aimed at target user groups. The web user access patterns calls for incorporation of machine intelligence techniques for mining meaningful information. The statistical web log data is generated by AWStats web access log file analyzer. HRFSOM clusters and analyzes information related to user access patterns from statistical data. HRFSOM is developed hierarchically by incorporating rough fuzzy set with SOM and has superior clustering accuracy. The cluster information is used by TSFIS variant to predict daily and hourly traffic. The results indicate cluster information significance to improve prediction accuracy of inference system. The experimental results demonstrate superiority of proposed method.

References

1. Jansen, B.J.: Understanding User-Web Interactions via Web analytics. 1st edn. Synthesis Lectures on Information Concepts, Retrieval and S. Morgan and Claypool Publishers (2009)
2. Chaudhuri, A.: Weblog Prediction with Machine Leaning Methods. Technical report, Samsung R&D Institute Delhi India (2016)

3. Clifton, B.: Advanced Web Metrics with Google Analytics. 3rd edn., Sybex (2012)
4. Yandex Personalized Web Search Challenge 2014. https://www.kaggle.com/c/yandex-personalized-web-search-challenge/data
5. AWStats web log file analyzer. http://www.awstats.org/
6. Kohonen, T.: Self-Organizing Map, 3rd Extended edn. Springer Series in Information Sciences, vol. 30. Springer, Heidelberg (2001)
7. Lingras, P.: Fuzzy rough and rough fuzzy serial combinations in neurocomputing. Neurocomputing. **36**(1), 29–44 (2001)
8. Pratihar, D.K.: Soft Computing: Fundamentals and Applications, 1st edn. Alpha Science International Ltd. (2013)

Approach to the Minimum Cost Flow Determining in Fuzzy Terms Considering Vitality Degree

Victor Kureichik and Evgeniya Gerasimenko(✉)

Southern Federal University, Taganrog, Russia
{vmkureychik, egerasimenko}@sfedu.ru

Abstract. An algorithm is presented to determine the minimum cost flow in a fuzzy network taking into account vitality degree. Algorithm consists in iteratively finding the paths of the minimum cost with vitality degrees no less than required one and pushing the flows along these paths. Network's parameters are presented in a fuzzy form due to the impact of environment factors and human activity. The proposed algorithm is based on the introduced rules of the residual network building. The numerical example is given that operated data from geoinformation system "ObjectLand" that contains information about railway system of Russian Federation. Initial data in a fuzzy form allow turning to the fuzzy graph with nodes presented by stations and arcs – by paths among them.

Keywords: Fuzzy network · Vitality · Fuzzy minimum cost flow

1 Introduction

Flow problems arise when commodities should be transported from the source to the sink in the network. Applications of such problems are railway and communication systems, logistics etc. Initially the first flow problem mentioned in the literature was the maximum flow problem, considered by Ford and Fulkerson within their famous labeling method. The second relevant problem arising while considering flows in networks was the minimum cost flow problem, which goal is to find the cheapest paths of transportation and pass the flow along them. R. Busacker and P. Gowen, M. Klein's [1] suggested methods for solving this task. There a lot of various modifications of such algorithms either in a form of linear programming [2] or graph techniques [3]. However, the majority of modifications are focused on the time improvement and don't deal with uncertainty existing in networks. The fact is that traditionally network's parameters include uncertainty at their specifying. Environment factors, human activity, the lack of information or its obsolescence influence arc capacities and arc costs [4]. Therefore, fuzzy logic should be used while specifying these factors.

Considered networks can include arcs with the profitability thresholds of transportation. Passing the flow along such arcs is expensive, therefore, one should transport certain amount of flow along these arcs that is defined by profitability threshold. Such constraints lead to necessity of checking the flow on these arcs in such a way, that the flow should be more than profitability threshold set in the form of the nonzero lower

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_20

flow bound. Thus, the condition of feasibility of flow reduces to finding the arcs with nonzero lower flow bounds [5, 6] and saturating them by the necessary flow value.

Another urgent task is considering vitality [7] providing network analysis. Vitality of the network is subjunctive assessment related to reliability of the system. In the present work vitality degree is presented as fuzzy number and set by experts according to such railway parameters, as life-time of railway segment and its length or station type and others. Considering vitality researchers can predict reliability of transportation, risks connected with emergencies, find roads that should be repaired.

Summarizing foresaid, the task considered in the present paper is the task of the minimum cost flow finding in fuzzy network with nonzero lower flow bounds and vitality degree.

The following paper has the following structure: Sect. 2 presents proposed method of the minimum cost task solving with vitalities. Practical realization and numerical example are given in Sect. 3. Section 4 is conclusion and future work.

2 Proposed Method of the Minimum Cost Flow Finding in the Fuzzy Network, Considering Vitality Degree

Consider definition of the fuzzy transportation network.

Fuzzy transportation network is a fuzzy network [8], presented as fuzzy directed graph $\tilde{G} = (X, \tilde{A})$, where $X = \{x_1, x_2, \ldots, x_n\}$ is the node set, $\tilde{A} = \{<\mu_{\tilde{A}}<x_i, x_j> / <x_i, x_j>\}$, $<x_i, x_j> \in X^2$, $\mu_{\tilde{A}}<x_i, x_j>$ – the fuzzy set of the arcs, where $\mu_{\tilde{A}}(x_i, x_j)$ is a grade of membership of the directed arc $<x_i, x_j>$ to the fuzzy set of the directed arcs $\tilde{A}$. Fuzzy arc capacity is applied as $\mu_{\tilde{A}}(x_i, x_j)$.

Let's give a problem statement of the minimum cost flow finding in the fuzzy network, considering vitality.

$$\sum_{(x_i, x_j) \in \tilde{A}^*} \tilde{c}_{ij} \tilde{\xi}_{ij} \to \min, \tag{1}$$

$$\sum_{x_j \in \Gamma(x_i)} \xi_{ij} = \sum_{x_k \in \Gamma^{-1}(x_i)} \xi_{ki} = \begin{cases} \rho, \; x_i = s, \\ -\rho, \; x_i = t, \\ 0, \; x_i \neq s, \, t, \end{cases} \tag{2}$$

$$l_{ij} \le \xi_{ij} \le u_{ij}, \; \forall \, (x_i, x_j) \in A. \tag{3}$$

Other words, it is necessary to transport given flow value in such a way that the cost of transportation would be minimal and the vitality of final flow would be no less than required.

In the model (1)–(3) $\tilde{A}^*_{ij}$ – the set of arcs, for which the condition $\tilde{v}_{ij} \ge \tilde{v}_{req}$ is valid, $\tilde{l}_{ij}$ – fuzzy lower flow bound for the arc (x_i, x_j), $\tilde{u}_{ij}$ – fuzzy upper flow bound for the arc (x_i, x_j), $\tilde{c}_{ij}$ – transmission cost of the one flow unit along the arc (x_i, x_j); $\tilde{\rho}$ – given flow value, which transmission cost should be minimized.

Proposed method operates fuzzy values of network parameters in a way that differs from conventional algorithms. Traditional methodic of fuzzy numbers adding and subtracting leads to "blurring" of the final value that is connected with risks and decreasing of effectiveness of fuzzy numbers calculations. Important feature of subtraction operation is that while subtracting of two equal fuzzy numbers we don't obtain zero fuzzy number that contradicts a condition of obtaining saturated arc after each flow distribution. Another significant item of conventional fuzzy subtraction operation consists in opportunity of negative result receiving. This fact contradicts either non-negativity condition. Consequently, variant method is required. Let us present it. Initially basic values of transmission costs will be set. We will operate central values of fuzzy numbers and for their deviations finding it should be referred to basic values of costs. Deviation borders of the final fuzzy number can be calculated as linear combination of the left and right borders of adjacent values.

$$\begin{aligned} l^L &= \frac{(a_2 - a')}{(a_2 - a_1)} \times l_1^L + \left(1 - \frac{(a_2 - a')}{(a_2 - a_1)}\right) \times l_2^L, \\ l^R &= \frac{(a_2 - a')}{(a_2 - a_1)} \times l_1^R + \left(1 - \frac{(a_2 - a')}{(a_2 - a_1)}\right) \times l_2^R. \end{aligned} \tag{4}$$

In (4) l^L is the left deviation border of the fuzzy triangular number with the center a'; l^R – the right deviation border of the fuzzy triangular number with the center a'. This method is described in [5, 8].

Proposed method is based o the following rules.

Rule 1 of initial fuzzy graph transformation to the graph without lower flow bounds

Transform initial graph $\tilde{G} = (X, \tilde{A})$ to the form of the graph without lower flow bounds $\tilde{G}^* = (X^*, \tilde{A}^*)$ introducing artificial nodes s^*, t^* and the reverse arc (t, s) with $\tilde{u}^*_{ts} = \infty$, $\tilde{l}^*_{ts} = \tilde{0}$, $\tilde{v}^*_{ts} = \tilde{1}$. For nodes $(x_i,\ x_j)$ in $\tilde{G}$ with nonzero lower flow bounds the following changes are valid: $\tilde{u}^*_{ij} = \tilde{u}_{ij} - \tilde{l}_{ij}, \tilde{l}_{ij} = \tilde{0}$, $\tilde{v}^*_{ij} = \tilde{v}_{ij}$ and artificial arcs $(s^*,\ x_j)$ and $(x_i,\ t^*)$ with flow bounds equal to $\tilde{u}^*_{s^*j} = \tilde{u}^*_{it^*} = \tilde{l}_{ij}$, $\tilde{l}^*_{s^*j} = \tilde{l}^*_{it^*} = \tilde{0}$, costs $\tilde{c}^*_{s^*x_j} = \tilde{c}^*_{x_it^*} = \tilde{0}$, $\tilde{c}^*_{s^*x_j} = \tilde{c}^*_{x_it^*} = \tilde{0}$, vitality parameters $\tilde{v}^*_{s^*j} = \tilde{v}^*_{it^*} = \tilde{1}$ should be introduced.

Rule 2 of construction of the fuzzy residual network $\tilde{G}^{*\mu} = (X^{*\mu}, \tilde{A}^{*\mu})$ for the minimum cost flow determining

Fuzzy residual network $\tilde{G}^{*\mu} = (X^{*\mu}, \tilde{A}^{*\mu})$ is a network, where $X^{*\mu} = X^*$, $\tilde{A}^{*\mu} = \{<\tilde{u}^{*\mu}_{ij}/(x^{*\mu}_i,\ x^{*\mu}_j)>\}$. It is built according to the following rules: for all arcs, if $\begin{cases} \tilde{\xi}^*_{ij} < \tilde{u}^*_{ij}, \\ \tilde{v}^*_{ij} \geq \tilde{v}_{req} \end{cases}$, then introduce corresponding arc in $\tilde{G}^{*\mu}$ with upper flow bounds $\tilde{u}^{*\mu}_{ij} = \tilde{u}^*_{ij} - \tilde{\xi}^*_{ij}$, costs $\tilde{c}^{*\mu}_{ij} = \tilde{c}^*_{ij}$.

For all arcs, if $\begin{cases} \tilde{\xi}^*_{ij} > \tilde{0}, \\ \tilde{v}^*_{ij} \geq \tilde{v}_{req} \end{cases}$,

then introduce corresponding arc in $\tilde{G}^{*\mu}$ with upper flow bounds $\tilde{u}^{*\mu}_{ji} = \tilde{\xi}^*_{ij}$, costs $\tilde{c}^{*\mu}_{ij} = -\tilde{c}^*_{ij}$.

Rule 3 of finding the graph with the feasible flow

Turn from $\tilde{G}^*$ to $\tilde{G}$ as follows: reject artificial nodes and arcs. Therefore, the feasible flow vector $\tilde{\xi} = (\tilde{\xi}_{ij})$ of the value $\tilde{\sigma}$ is defined as $\tilde{\xi}_{ij} = \tilde{\xi}^*_{ij} + \tilde{l}_{ij}$.

Represent algorithm for solving the minimum cost flow task with vitalities in fuzzy terms.

Algorithm for the minimum cost flow determining in the fuzzy network with the given vitality degree

Step 1. Determine the existence of the feasible flow in the network, presented as fuzzy graph $\tilde{G} = (X, \tilde{A})$ according to construction of the graph $\tilde{G}^* = (X^*, \tilde{A}^*)$ due to the *rule 1*.

Step 2. Let us build a fuzzy residual network $\tilde{G}^{*\mu}$ corresponding to the graph $\tilde{G}^* = (X^*, \tilde{A}^*)$ as in the *rule 2*.

Step 3. Obtain the minimum cost $\tilde{P}^{*\mu}$ from the artificial source s^* to the artificial sink t^* in $\tilde{G}^{*\mu}$.

3.1. If the $\tilde{P}^{*\mu}$ exists, turn to the **step 4.**

3.2. The value $\tilde{\phi} < \sum\limits_{\tilde{l}_{ij} \neq 0} \tilde{l}_{ij}$ is determined, if the path doesn't exist, i.e., one can't pass any flow units, however all artificial arcs are not saturated. Other words, there is o feasible flow in $\tilde{G}$.

Step 4. Push the value $\tilde{\delta}^{*\mu} = \min[\tilde{u}(\tilde{P}^{*\mu})]$, $\tilde{u}(\tilde{P}^{*\mu}) = \min[\tilde{u}^{*\mu}_{ij}]$, $(x^{*\mu}_i, x^{*\mu}_j) \in \tilde{P}^{*\mu}$ along the minimum cost path $\tilde{P}^{*\mu}$.

Step 5. Reset the flow values in $\tilde{G}^*$: reset the flows $\tilde{\xi}^*_{ji}$ by $\tilde{\xi}^*_{ji} - \tilde{\delta}^{*\mu}$ for the arcs $(x^{*\mu}_i, x^{*\mu}_j) \notin \tilde{A}^*$, $(x^{*\mu}_i, x^{*\mu}_j) \in \tilde{A}^{*\mu}$ in $\tilde{G}^{*\mu}$. Reset the flows $\tilde{\xi}^*_{ij}$ by $\tilde{\xi}^*_{ij} + \tilde{\delta}^{*\mu}\tilde{P}^{*\mu}$ for the arcs $(x^{*\mu}_i, x^{*\mu}_j) \in \tilde{A}^*$, $(x^{*\mu}_i, x^{*\mu}_j) \in \tilde{A}^{*\mu}$ in $\tilde{G}^{*\mu}$

Step 6. Match the values of $\tilde{\xi}^*_{ij} + \tilde{\delta}^{*\mu}\tilde{P}^{*\mu}$ **and** $\sum\limits_{\tilde{l}_{ij} \neq \tilde{0}} \tilde{l}_{ij}$:

6.1. If the condition $\tilde{\sigma}^* < \sum\limits_{\tilde{l}_{ij} \neq \tilde{0}} \tilde{l}_{ij} \leq \tilde{\rho}$ is valid, where $\tilde{\sigma}^*$ is the value $\tilde{\sigma}^* = \tilde{\xi}^*_{ij} + \tilde{\delta}^{*\mu}\tilde{P}^{*\mu}$ of the minimum cost $\tilde{c}(\tilde{\xi}^*_{ij} + \tilde{\delta}^{*\mu}\tilde{P}^{*\mu})$, go to the **step 2**.

6.2. If the condition $\tilde{\sigma}^* = \sum\limits_{\tilde{l}_{ij} \neq \tilde{0}} \tilde{l}_{ij} \leq \tilde{\rho}$ is true, therefore, the value $\tilde{\xi}^*_{ij} + \tilde{\delta}^{*\mu}\tilde{P}^{*\mu}$ of the minimum cost $\tilde{c}(\tilde{\xi}^*_{ij} + \tilde{\delta}^{*\mu}\tilde{P}^{*\mu})$ is found. One can come to the conclusion that the flow along the artificial arc (t, s) in $\tilde{G}^*$ is the feasible flow in the initial graph $\tilde{G}$ of the value $\tilde{\sigma} = \tilde{\xi}^*_{ts}$. Go to the graph $\tilde{G}$ from the graph $\tilde{G}^*$ according to the *rule 3*. The network $\tilde{G}(\tilde{\xi})$ with the feasible is defined. Determine, if the flow is optimal.

6.2.1. If the flow in $\tilde{G}(\tilde{\xi})$ is equal to $\tilde{\rho}$ of the cost $\sum\limits_{(x_i, x_j) \in \tilde{A}} \tilde{c}_{ij}\tilde{\xi}_{ij}$, we find the minimum cost flow, **the end**.

6.2.2. If the flow in $\tilde{G}(\tilde{\xi})$ is less than $\tilde{\rho}$ of the cost $\sum_{(x_i,x_j)\in\tilde{A}} \tilde{c}_{ij}\tilde{\xi}_{ij}$, we obtain the network $\tilde{G}(\tilde{\xi})$ with the feasible flow. Turn to the **step 7.**

Step 7. Build the residual network $\tilde{G}^{\mu}(\tilde{\xi})$ by the feasible flow vector $\tilde{\xi} = (\tilde{\xi}_{ij})$ in the graph $\tilde{G}$.

Step 8. Determine the shortest path $\tilde{P}^{\mu}$ in $\tilde{G}^{\mu}(\tilde{\xi})$.

8.1. Turn to the **step 9** if the shortest path $\tilde{P}^{\mu}$ is found.

8.2. If there is no such a path, then $\tilde{\rho} > \tilde{v}$ and the task has no solution, **the end.**

Step 9. Push $\tilde{\delta}^{\mu} = \min[\tilde{u}(\tilde{P}^{\mu})]$, $\tilde{u}(\tilde{P}^{\mu}) = \min[\tilde{u}^{\mu}_{ij}]$, $(x^{\mu}_i, x^{\mu}_j) \in \tilde{P}^{\mu}$ along the shortest path.

Step 10. Reset the values of flow in $\tilde{G}$: substitute the fuzzy flow $\tilde{\xi}_{ji}$ by $\tilde{\xi}_{ji} - \tilde{\delta}^{\mu}$ for arcs $(x^{\mu}_i, x^{\mu}_j) \notin \tilde{A}$, $(x^{\mu}_i, x^{\mu}_j) \in \tilde{A}^{\mu}$ in $\tilde{G}^{\mu}(\tilde{\xi})$ and the fuzzy flow $\tilde{\xi}_{ij}$ by $\tilde{\xi}_{ij} + \tilde{\delta}^{\mu}$ for arcs $(x^{\mu}_i, x^{\mu}_j) \in \tilde{A}$, $(x^{\mu}_i, x^{\mu}_j) \in \tilde{A}^{\mu}$ in $\tilde{G}^{\mu}(\tilde{\xi})$. Finally, substitute the flow value $\tilde{\xi}_{ij}$ by $\tilde{\xi}_{ij} + \tilde{\delta}^{\mu}\tilde{P}^{\mu}$ in $\tilde{G}$ and go to the **step 11**.

Step 11. Match the value $\tilde{\xi}_{ij} + \tilde{\delta}^{\mu}\tilde{P}^{\mu}$ and $\tilde{\rho}$:

11.1. If the condition $\tilde{\xi}_{ij} + \tilde{\delta}^{\mu}\tilde{P}^{\mu} < \tilde{\rho}$ is satisfied, then substitute $\tilde{\xi}_{ij}$ by $\tilde{\xi}_{ij} + \tilde{\delta}^{\mu}\tilde{P}^{\mu}$ and go to the **step 7**.

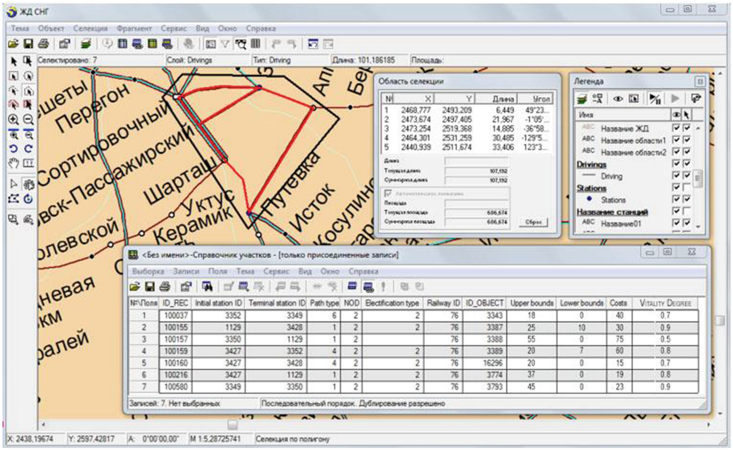
11.2. If the condition $\tilde{\xi}_{ij} + \tilde{\delta}^{\mu}\tilde{P}^{\mu} = \tilde{\rho}$ is satisfied, therefore, the required flow value of the minimum cost is obtained, **the end.**

11.3. If the condition $\tilde{\xi}_{ij} + \tilde{\delta}^{\mu}\tilde{P}^{\mu} > \tilde{\rho}$ is satisfied, where flow value $\tilde{\xi}_{ij} + \tilde{\delta}^{\mu}\tilde{P}^{\mu} = \tilde{h}$, therefore, the required flow is $\tilde{\xi}_{ij} + (\tilde{\delta}^{\mu} - \tilde{h} + \tilde{\rho})\tilde{P}^{\mu}$ of the minimum cost $\tilde{c}(\tilde{\xi}_{ij} + (\tilde{\delta}^{\mu} - \tilde{h} + \tilde{\rho})\tilde{P}^{\mu})$, **the end.**

3 Numerical Example

Present numerical example, illustrating the proposed method. The required data for the task are taken from the geoinformation system «ObjectLand» 2.6 [9]. The system contains information about railway system of Russian Federation. User can select necessary part of the map and obtain information about stations, paths etc. It is possible either to select a segment of the railway map and turn to representation in the form of the node-graph, where stations names are the nodes of selected graph, as shown in Fig. 1. The task is to calculate the minimum transportation cost of $\tilde{30}$ flow units considering nonzero lower flow bounds and given vitality degrees in such a way that the final flow would satisfy required vitality constraint – required vitality degree is no less than $0.\tilde{7}$ units.

Fuzzy parameters of the network, such as flow bounds, costs and vitality degrees are set by experts and incorporated in the GIS in the forms of tables. Particular flow task can be solved according to this selection. Consider the solution of the minimum cost flow problem taken into account vitality degrees in fuzzy terms. Represent selected area as fuzzy graph, as presented in Fig. 2.



Область селекции

N	X	Y	Длина	Угол
1	2468,777	2493,209	6,449	49°23...
2	2473,674	2497,405	21,967	-1°05'...
3	2473,254	2519,368	14,885	-36°58...
4	2464,301	2531,259	30,485	-129°5...
5	2440,939	2511,674	33,406	123°3...

Длина
Текущая длина 107,192
Суммарная длина 107,192
☑ Автоматическое замыкание
Площадь
Текущая площадь 686,674
Суммарная площадь 686,674

<Без имени>-Справочник участков - [только присоединенные записи]

№\Поля	ID_REC	Initial station ID	Terminal station ID	Path type	NOD	Electrification type	Railway ID	ID_OBJECT	Upper bounds	Lower bounds	Costs	Vitality Degree
1	100037	3352	3349	6	2	2	76	3343	18	0	40	0.7
2	100155	1129	3428	1	2	2	76	3387	25	10	30	0.9
3	100157	3350	1129	1	2		76	3388	55	0	75	0.5
4	100159	3427	3352	4	2	2	76	3389	20	7	60	0.8
5	100160	3427	3428	4	2	2	76	16296	20	0	15	0.7
6	100216	3427	1129	1	2	2	76	3774	37	0	19	0.8
7	100580	3349	3350	1	2	2	76	3793	45	0	23	0.9

Записей: 7. Нет выбранных | Последовательный порядок. Дублирование разрешено

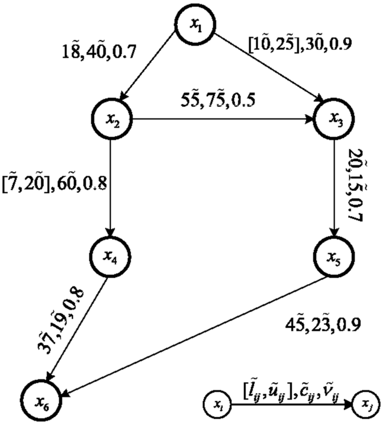
X: 2438,19674 | Y: 2597,42817 | A: 0°00'00,00" | M 1:5,28725741 | Селекция по полигону

Fig. 1. Data in GIS "ObjectLand"

Turn to the network without lower flow bounds, as presented in Fig. 3. We add artificial nodes and arcs, connecting them with nodes according to the rule 1.

Define if the initial network has a feasible flow trying to pass the maximum flow units equal to the sum of the lower flow bounds in the residual network. Note, that the flow can be pushed along the arcs with the value of vitality not less than given vitality value. At the first step residual network coincides with the graph without lower flow bounds and constructed by the rule 2.

Find the minimum cost path s^*, x_3, x_5, x_6, x_1, t^* of the cost $3\tilde{8}$ conventional units and pass $1\tilde{0}$ flow units along it.



x_1
$\tilde{18},\tilde{40},0.7$
$[\tilde{10},\tilde{25}],\tilde{30},0.9$
x_2
$\tilde{55},\tilde{75},0.5$
x_3
$[\tilde{7},\tilde{20}],\tilde{60},0.8$
$\tilde{20},\tilde{15},0.7$
x_4
x_5
$\tilde{37},\tilde{19},0.8$
$\tilde{45},\tilde{23},0.9$
x_6
x_i
$[\tilde{l}_{ij},\tilde{u}_{ij}],\tilde{c}_{ij},\tilde{v}_{ij}$
x_j

Fig. 2. Initial fuzzy graph

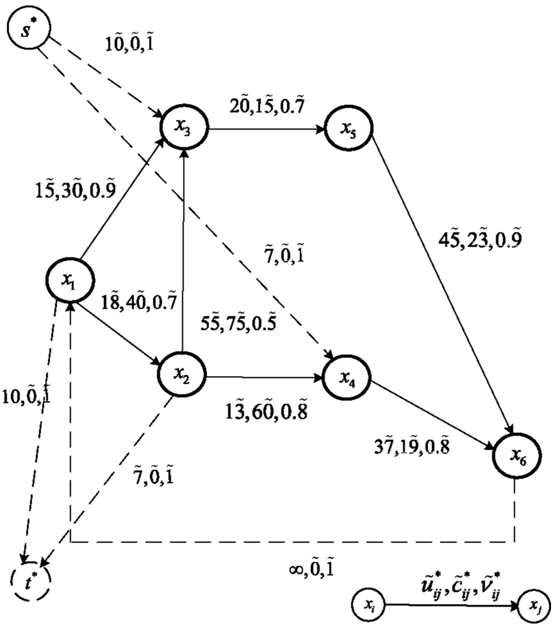

s^*
$\tilde{10},\tilde{0},\tilde{1}$
$\tilde{20},\tilde{15},0.\tilde{7}$
x_3
x_5
$\tilde{15},\tilde{30},0.\tilde{9}$
$\tilde{45},\tilde{23},0.\tilde{9}$
$\tilde{7},\tilde{0},\tilde{1}$
x_1
$\tilde{18},\tilde{40},0.\tilde{7}$
$\tilde{55},\tilde{75},0.\tilde{5}$
$10,\tilde{0},\tilde{1}$
x_2
x_4
$\tilde{13},\tilde{60},0.\tilde{8}$
$\tilde{37},\tilde{19},0.\tilde{8}$
x_6
$\tilde{7},\tilde{0},\tilde{1}$
$\infty,\tilde{0},\tilde{1}$
t^*
x_i
$\tilde{u}^*_{ij},\tilde{c}^*_{ij},\tilde{v}^*_{ij}$
x_j

Fig. 3. Fuzzy graph without lower flow bounds

Then construct residual network according to recent changes and search the minimum cost path in it: s^*, x_4, x_6, x_1, x_2, t^* of the cost $\tilde{59}$ conventional units and pass $\tilde{7}$ flow units along this path.

Arcs that are incident to the source and sink are saturated, therefore, we find the maximum flow in this graph. The network without lower flow bounds with the maximum flow value is represented in Fig. 4. As the flow in the network is equal to the sum of the lower flow bounds, the feasible flow in the initial network exists and is equal to the flow, passing along the artificial arc *(t,s)*.

After that we can turn to the graph construction with the feasible flow according to the rule 3. Delete all artificial nodes and arcs and change flow values along the arcs and construct graph that presented in Fig. 5.

As obtained feasible flow of the value $\tilde{17}$ units is less than required flow value, build a fuzzy residual network of this graph, as shown in Fig. 6. Search the minimum cost paths in this network with further iteratively passing the flows along them before required value obtaining.

The first path of the minimum cost is: x_1, x_3, x_5, x_6 of the cost $\tilde{68}$ conventional units, pass $\tilde{10}$ flow units along it. At the next step search the minimum cost path in the updated network: x_1, x_2, x_4, x_6 of the cost $\tilde{119}$ conventional units, pass $\tilde{3}$ flow units along it.

Finally, we come to the fuzzy graph with the flow value of $\tilde{30}$ units that is given flow value, as shown in Fig. 7. Then, stop. Calculate minimal transmission cost of the $\tilde{30}$ flow units: $\tilde{20} \times (\tilde{30} + \tilde{15} + \tilde{23}) + \tilde{10} \times (\tilde{40} + \tilde{60} + \tilde{19}) = \tilde{2550}$ conventional units.

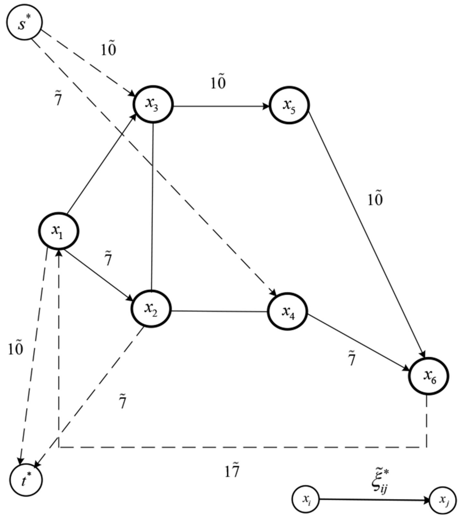


Fig. 4. Graph without lower flow bounds with the maximum flow value

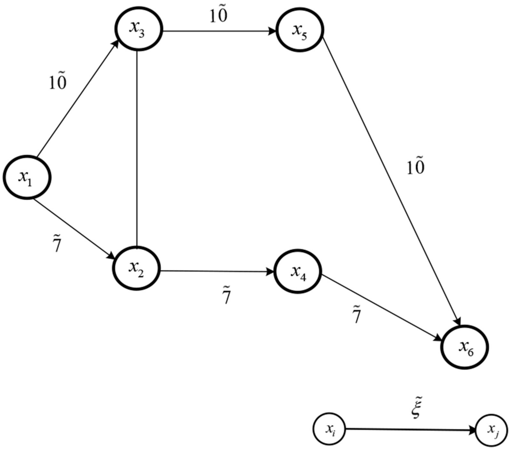


Fig. 5. Initial fuzzy graph with the feasible flow

According to Fig. 8 this arc cost value is between neighboring values $214\tilde{0}$ with the left deviation $l^L = 190$, right deviation $l^R = 220$ and $304\tilde{0}$ with the left deviation $l^L = 260$, right deviation $l^R = 280$. According to Eq. (4) obtain deviations of this number: $l^L = 236$ and $l^R = 260$. Therefore, total minimum cost of transportation is represented as triangular number (2314, 2550, 2810) conventional units.

Time complexity of the task is defined by the choice of the minimum cost path algorithm and varies from $O(X^2 + logX + XA)$ in the case of Jonson's algorithm to $O(X^3)$ while using Floyd-Warshall's algorithm.

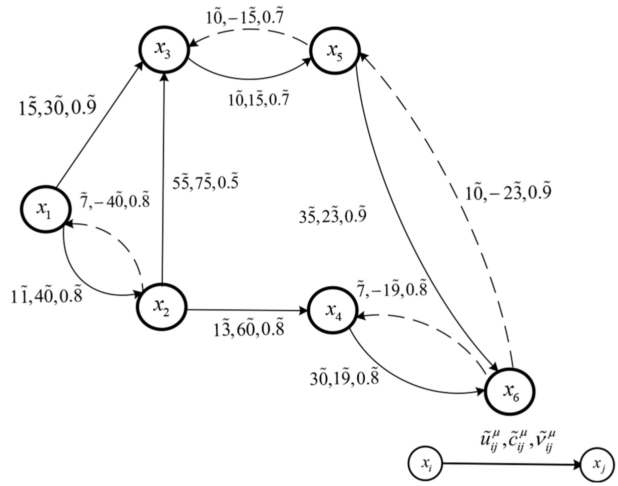

$\tilde{10},-\tilde{15},0.\tilde{7}$
x_3
x_5
$\tilde{15},\tilde{30},0.\tilde{9}$
$\tilde{10},\tilde{15},0.\tilde{7}$
$\tilde{55},\tilde{75},0.\tilde{5}$
x_1
$\tilde{7},-\tilde{40},0.\tilde{8}$
$\tilde{10},-\tilde{23},0.\tilde{9}$
$\tilde{35},\tilde{23},0.\tilde{9}$
$\tilde{11},\tilde{40},0.\tilde{8}$
x_2
$\tilde{7},-\tilde{19},0.\tilde{8}$
x_4
$\tilde{13},\tilde{60},0.\tilde{8}$
$\tilde{30},\tilde{19},0.\tilde{8}$
x_6
$\tilde{u}_{ij}^{\mu},\tilde{c}_{ij}^{\mu},\tilde{v}_{ij}^{\mu}$
x_i
x_j

Fig. 6. Residual network for the graph with the feasible flow

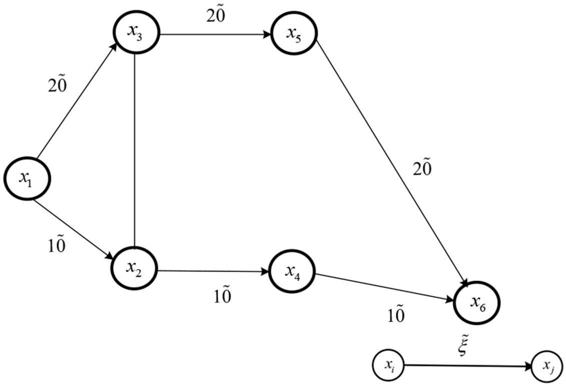


Fig. 7. Fuzzy graph for $\tilde{30}$ flow units

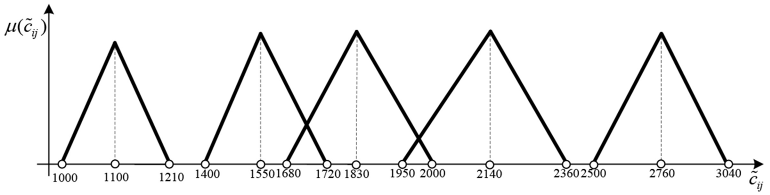


Fig. 8. Basic values of arc costs

The proposed algorithm is optimal, because the cheapest path is found at each step of the method. Besides, it is able to work with lower software resources.

4 Conclusion and Future Work

Method of the minimum cost flow finding in fuzzy terms with given vitality degrees is presented in the paper, based on the proposed rules of residual network construction. Proposed method takes into account fuzziness peculiar to network's parameters, as environment changes, human activity, errors in measurements influence arc capacities and transmission costs. Nonzero lower flow bounds allow considering profitability of transportation, given vitality degrees and required vitality reflect reliability of transportation. Numerical example is presented based on GIS «ObjectLand» . Considered method has important practical value, because researchers can solve real optimization tasks, including minimum cost flow determining on real railway roads. In the future works methods of two-commodity flow finding will be developed considering vitality degree.

Acknowledgments. This work has been supported by the Russian Foundation for Basic Research, Project № 16-01-00090 a.

References

1. Busacker, R.G., Gowen, P.: A procedure for determining a family of minimum-cost network flow patterns. Technical report 15, Operations Research Office, John Hopkins University (1961)
2. Ganesan, K., Veeramani, P.: Fuzzy linear programs with trapezoidal fuzzy numbers. Ann Oper. Res. **143**, 305–315 (2006)
3. Kovács, P.: Minimum-cost flow algorithms: An experimental evaluation. EGRES Technical report, № 4 (2013)
4. Bozhenyuk, A.V., Rozenberg, I.N., Rogushina, E.M.: Approach of maximum flow determining for fuzzy transportation network. Izvestiya SFedU. Eng. Sci. **5**(118), 83–88 (2011)
5. Bozhenyuk, A., Gerasimenko, E.: Flows finding in networks in fuzzy conditions. In: Kahraman, C., Öztayşi, B. (eds.) Supply Chain Management Under Fuzziness. STUDFUZZ, vol. 313, pp. 269–291. Springer, Heidelberg (2014). doi:10.1007/978-3-642-53939-8_12
6. Yi, T., Murty, K.G.: Finding Maximum Flows in Networks with Nonzero Lower Bounds Using Preflow Methods. Technical report, IOE Dept., University of Michigan, Ann Arbor, Mich., (1991)
7. Bozhenyuk A.V., Gerasimenko E.M.: Maximum dynamic flow finding task with the given vitality degree. In: Proceedings of the 2nd International Workshop on Soft Computing Applications and Knowledge Discovery (SCAKD 2016), Moscow, Russia, 18 July 2016, pp. 75–86 (2016)
8. Belyakov, S.L., Bozhenyuk, A.V., Ginis, L.A., Gerasimenko, E.M.: Fuzzy Methods of Control by Streams in Geoinformation Systems. Southern Federal University, Taganrog (2013)
9. ObjectLand/Geoinformation system. http://www.objectland.ru/

Query Focused Multi document Summarization Based on the Multi facility Location Problem

Ercan Canhasi(✉)

Gjirafa, Inc., Rr. Rexhep Mala, 28A, Prishtine, Kosovo
ercan@gjirafa.com
http://www.gjirafa.com

Abstract. In this paper we propose a query focused multi document summarization method based on facility location problem. In order to formalize the sentence selection/extraction as a multi facility location problem (mFLP) we modeled the input documents as a sentence dissimilarity graph and a given query as a query to sentences similarity vector. In mFLP terminology the former is known as a cost to serve matrix, and latter as a cost to establish vector. By formulating the mFLP as the mixed integer linear programming problem we were able to optimally select sentences (facilities) which minimize the weighted sum of distances from each demand point to its nearest facility, plus the sum of opening costs of the facilities (query to sentences similarity). The performance of this new method has been tested using the DUC2005 and DUC2006 data corpus. The effectiveness of this technique is measured using the ROUGE score. The results indicate that presented methodology is truly a promising research direction.

Keywords: Query focused multi-document summarization · Facility location problem · Optimization · Mixed integer programming

1 Introduction

The massive quantity of online information has largely raised need for text summarization systems. Text summarization is an approach of automatically producing a brief version of a given text that provides useful information. The problem of extracting the most important sentences from text and the problem of generating rational summaries are the most important tasks of text summarization. There are two generic class of summarization: extractive and abstractive. Extractive summarization methods, as their name suggests, extensively reduce the complexity of the summarization by simply selecting the most characteristic subset of the given sentences. Contrary, abstractive summarization methods are required to compose new sentences that are not seen in the original documents. Based on the purpose, the summaries can be categorized into general and query-based summaries.

Query-focused multi-document summarization is a special case of multi-document summarization. Given a query, the task is to produce a summary which can respond to the information required by the query. Different from general [1, 2] summarization,

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_21

which needs to preserve the typical semantic essence of the original document(s) query-focused [3] summarization purposely demands the most typical summary biased toward an explicit query.

Since query focused MDS can produce brief information corresponding to the users queries, it can be applied to various tasks for satisfying different user interests. Queries are mostly real-world complex questions (e.g., “Track the spread of the West Nile virus through the United States.” is a query example). Such complicated questions make the query focused summarization task quite difficult. The real problem is how to model the question jointly with the documents to be summarized and thus bias the answer, i.e. summary, towards the provided question. Besides there are very few, the most existing research on applying optimization approaches to Q-MDS explores formulations such as knapsack problem or more general as a global inference problem [4], maximum coverage problem [5, 6], maximum coverage problem with knapsack constraint [7], multi objective optimization [8], and 0–1 nonlinear programming [9]. In the paper, we try to utilize a simple optimization method and study a new setup of the problem of query-focused summarization. Since the multi facility location problem [10] assembles the advantages of clustering and implementation simplicity we propose using the mFLP in Q-MDS. Consequently, the main concerns of the paper are: (1) how to incorporate query information in its own nature of a multi facility location based summarizer; and (2) how to increase the variability and diversity of the produced query-focused summary.

The main contributions of the paper are three-fold:

1. A novel query-focused summarization method mFLSum is proposed.
2. Modeling the input documents and query information as a sentence to sentence similarity graph (cost to serve matrix) and query to sentences similarity vector (cost to establish vector) is introduced.
3. The effectiveness of the proposed approach is examined in the context of Q-MDS.

To show the efficiency of the proposed approach, we compare it to other closely related summarization methods. We have used the DUC2005 and DUC2006 data sets to test our proposed method empirically. Experimental results show that our approach significantly outperforms the baseline summarization methods and well compares with other optimization approaches. The remainder of the paper is organized as follows: Sect. 2 describes related work. In Sect. 3 the multi facility location problem is introduced. The details of the proposed summarization approach mFLSum are also presented in Sect. 3, where we give an overview of the new approach and an illustrative example of its use. Section 4 shows the evaluation and experimental results. Finally, we conclude in Sect. 5.

2 Related Work

Multi document summarization systems, especially query sensitive ones, based on optimization frameworks have not been researched well enough. This is mainly by reason of struggle in formulating the criteria for objective evaluation. Commonly, the Greedy Maximum Marginal Relevance (MMR) [11] type methods were employed to treat significance and redundancy in a linear combination way. Summaries were created in an approximate greedy procedure that incrementally included the sentences that maximize the combined criteria. However, local greedy algorithms rarely found the best summary. The optimization based summarization approach presented in [4] is formulated as a global inference problem. In it, three properties are tried to optimize jointly, i.e., relevance (or significance), redundancy and length. Few different algorithms were introduced and evaluated, including a greedy approximate method, a dynamic programming approach based o solutions to the knapsack problem, and an exact algorithm that used an integer linear programming formulation of the problem. Meanwhile, Yih et al. [6] evaluated summaries according to the information coverage measured by content bearing words. In their work, each term in the documents was assigned a score calculated by using its frequency and position information. Then, an algorithm based on stack decoder was developed to search for the best combination of the sentences that maximized the sum of the scores subject to the length constraint. However, they did not explicitly assess information redundancy in their optimization model.

As far as we know, the idea of optimizing summarization was mentioned as early as 2004 in [5]. They represented documents in a two dimensional space of textual and conceptual units with an associated mapping between them, and proposed a formal model that simultaneously selected important text units and minimized information overlap between them. The selection of the best textual units was regarded as an optimization problem over a general scoring function that maximized the distinct conceptual units. The model was comprehensive, but it required further studies in order to practically specify and implement the abstract model.

Different from previous work which focused on single objective optimization, we explicitly define multiple summary evaluation criteria and formulate them as separate objective functions which are measured based on query sensitive core terms and main topics built from core term clusters.

Extractive document summarization presented in [9] is modeled as a 0–1 programming problem. The problem is formulated with taking into account three basic requirements, namely content coverage, diversity and length limit that should satisfy summaries. We defined an objective function as a weighted linear combination of the arithmetic and geometric means of the objective functions enforcing the coverage and diversity.

3 Multi facility Location Problem Based Document Summarization

In this section, we first present an overview of the multi facility location problem (mFLP), following with detailed multi document summarization (MDS) problem statement and a new summarization method, which employs mFLP for query oriented MDS. An illustrative example, discussions, and properties of the proposed method are also given.

3.1 Multi facility Location Problem

The location analysis, also known as facility location problem is a topic of computational geometry and operations research involved with the optimal installation of facilities to minimize some cost functions such as transportation while considering various dependent factors. In a multi facility location problem, another version of original facility location problem, the number of new facilities (NFs) to be established can either be specified or can be determined as part of the location algorithm. When the number of NFs is specified, the allocation of existing facilities EFs to NFs can either be given or determined as part of what is then termed a location–allocation problem. If the NF-to-EF allocations are given and there are no interactions between the NFs, then the multifacility problem reduces to a series of single-facility location problems. The location–allocation problem remains difficult even when there are no interactions between the NFs because of the need to determine the allocations. In this work the number of NFs is determined by algorithm and since the summarization method is extractive, selected sentences should be subset of existing ones. Hence, NFs are algorithmically selected from EFs.

The uncapacitated facility location problem can be defined as follows: Given *m* number of *EFs* and *n* sites at which *NFs* can be established, the uncapacitated facility location (UFL) problem can be formulated as the following mixed-integer linear programming (MILP) problem:

$$\min TC = \sum_{i=1}^{n} k_i y_i + \sum_{i=1}^{n} \sum_{j=1}^{ma} c_{ij} x_{ij} \tag{1}$$

subject to

$$\sum_{i=1}^{n} x_{ij} = 1, \quad j = 1, \ldots, m \tag{2}$$

$$y_i \geq x_{ij}, \quad i = 1, \ldots, n; \quad j = 1, \ldots, m \tag{3}$$

$$0 \leq x_{ij} \leq 1, \quad i = 1, \ldots, n; \quad j = 1, \ldots, m \tag{4}$$

$$y_i \in \{0, 1\}, \quad i = 1, \ldots, n; \tag{5}$$

Where

$k_i =$	fixed costs of establishing a NF at site i
$c_{ij} =$	variable costs to serve all of EFj's demand from site i
$y_1 = 1,$	if NF established at site *i*; 0, otherwise
$x_{ij} =$	fraction of EF j's demand served from NF at site i

The UFL problem is a MILP because the y_i's are binary variables and the x_{ij}'s are real variables. In the UFL problem, all x_{ij} are 0 or 1; in the capacitated facility location (CFL) problem, there is a maximum capacity associated with each site, resulting in a x_{ij}

value between 0 and 1 whenever not all of an EF j's demand can be served from the NF at site j.

3.2 MDS Problem Statement and Corpus Modeling

To utilize the mFLP for sentence extraction we use the model of sentence similarity matrix. Let a document corpus be separated into a set of sentences $D = \{s_1, s_2, \ldots, s_n\}$, where n denotes the number of sentences, and s_i denotes i_{th} sentence in D. In the interest of forming the term-sentence and sentence similarity matrices, each of the sentences should be presented as a vector. The vector space model is the most known representation scheme for textual units. It represents textual units by counting terms or sequence of terms. Let $T = \{t_1, t_2, \ldots, t_m\}$ represent all the distinct terms occurring in the collection, where m is the number of different terms. The standard vector space model (VSM) [12] using the bag of the words approach represents the text units of a corpus as vectors in a vector space. Traditionally, a whole document is used as a text unit, but in this work, we use only sentences. Each dimension of a vector corresponds to a term that is present in the corpus. A term might be, for example, a single word, N-gram, or a phrase. If a term occurs in a sentence, the value of that dimension is nonzero. Values can be binary, frequencies of terms in the sentence, or term weights. Term weighting is used to weight a term based on some kind of importance. The most often used measure is the raw frequency of a term, which only states how often the term occurs in a document without measuring the importance of that term within the sentence or within the whole collection. Different weighting schemes are available. The most common and popular one is the term frequency–inverse sentence frequency (tf-isf) weighting scheme. It combines local and global weighting of a term. The local term weighting measures the significance of a term within a sentence:

$$\mathrm{tf}_{ik} = \mathrm{freq}_{ik}$$

here freq_{ik} is the frequency of term t_k in sentence s_i. With this formula, terms that occur often in a sentence are assessed with a higher weight. The global term weighting or the inverse sentence frequency is f measures the importance of a term within the sentence collection:

$$\mathrm{isf}_{ik} = \log\left(\frac{n}{n_k}\right)$$

where n denotes the number of all sentences in the corpus, and n_k denotes the number of sentences that term t_k occurs in. This formula gives a lower isf value to a term that occurs in many sentences, and in this way, it favors only the rare terms since they are significant for the distinction between sentences. As a result, the tf-isf weighting scheme can be formulated as the following:

$$w_{ik} = tf_{ik} \times isf_{ik} = freq_{ik} \times \log\left(\frac{n}{n_k}\right)$$

where the weight w_{ik} of a term t_k in a sentence s_i is defined by the product of the local weight of term t_k in sentence s_i and the global weight of term t_k. A very popular similarity measure is the cosine similarity, which uses the weighting terms representation of the sentences. According to the VSM, the sentence si is represented as a weighting vector of the terms, $s_i = [w_{i1}, w_{i2}, \ldots, w_{im}]$, where w_{ik} is the weight of the term tk in the sentence si. This measure is based on the angle α between two vectors in the VSM. The closer the vectors are to each other the more similar are the sentences. The calculation of an angle between two vector $s_i = [w_{i1}, w_{i2}, \ldots, w_{im}]$ and $s_j = [w_{j1}, w_{j2}, \ldots, w_{jm}]$ can be derived from the Euclidean dot product:

$$s_i, s_j = |s_i| \cdot |s_j| \cdot \cos\alpha$$

This states that the product of two vectors is given by the product of their norms (in spatial terms, the length of the vector) multiplied by the cosine of the angle α between them. Given

Equation 7, the cosine similarity is therefore

$$\operatorname{sim} s_i, s_j = \cos\alpha = \frac{s_i, s_j}{|s_i|} \cdot |s_j|$$

$$= \frac{\sum_{l=1}^{m} w_{il} w_{jl}}{\sqrt{\sum_{l=1}^{m} w_{il}^2 \sum_{l=1}^{m} w_{jl}^2}}, \quad i, j = 1, 2, \ldots, n.$$

The sentence similarity matrix describes a similarity between sentences presented as points in Euclidean space. Columns and rows are sentences, while their intersection gives the similarity values of corresponding sentences calculated with Eq. 8.

3.3 Query-Focused Document Summarization

In this section, a method for generating the query focused multi document summary by selecting sentences using the mFL is presented. We give an overall illustration of the method in Fig. 1. The main idea of the method is simple: sentences are assigned to some "centers" fined by mFL algorithm in order to produce the sentence ranking where the top ranked ones are then sequentially extracted, until the length constraint (l sentences) is reached.

The framework of the proposed method, mFLSum, consists of the following steps:

1. Construct the input matrix C using Eq. (5).
2. Construct the input vector y using Eq. (6).
3. Perform mFL on matrix X given the vector y.
   (a) Estimate the output matrix X and vector y as described in Sect. 3.2
   (b) Calculate values of $\boldsymbol{O_i}$; which are the sums of the served EFs by NFs
   (c) order NFs in descending order of sums.

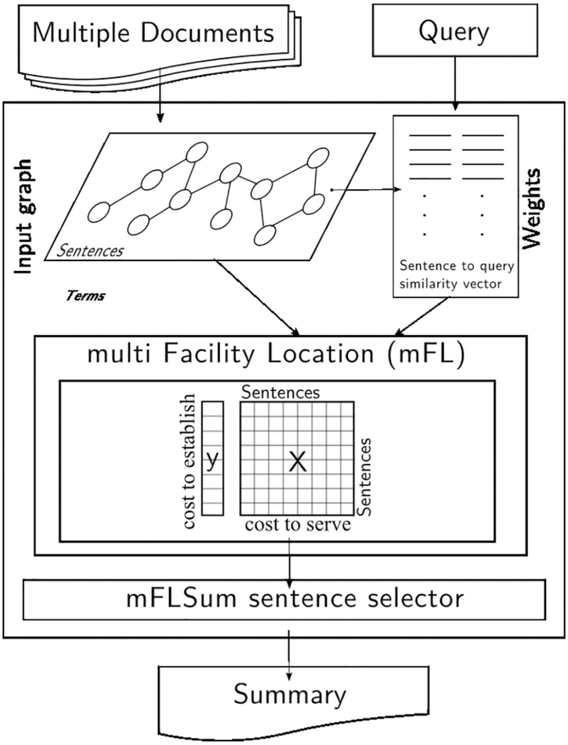


Fig. 1. Query focused MDS method using mFL

4. Select k sentences with the highest O_i values.
   (a) Start with the most significant NF and extract sentences in the order according to their values in $\boldsymbol{O_i}$. That is, sentences with highest number of serving existing facilities from each new facility are selected and if the summary length is not met then the extraction step continues with the second highest values, and so on.
   (b) Each selected sentence is compared to previously selected ones and if there is a significant similarity between them, the newly selected sentence is not included in the summary.

In the above algorithm, the third and fourth steps are crucial. Our purpose is to assign sentences into new locations and subsequently select the sentences (NFs) with the highest ordering values O_i. In the third step, the significant sentences are more likely to be assigned to NF with high significance. Since the sentences with the higher location ordering are ranked higher, the sentences selected by the fourth step are the most central ones. The fourth step in the above algorithm starts the sentence extraction with the largest NF to ensure that the system-generated summary first covers the facts that have higher weights. In this way our method optimizes the two important aspects of the summarization, namely the relevance and the content coverage. The last important effect of these two steps is diversity optimization. In order to effectively remove redundancy and increase the information diversity in the summary, we use a greedy algorithm presented in the last step (4.ii) of above algorithm. In the following subsection we present the usage of mFLSum on an illustrative example.

4 Experiments

In this section, we conduct experiments on two DUC data sets to evaluate the effectiveness and possible positive contributions of the proposed method compared with other existing summarization systems.

4.1 Experimental Data and Evaluation Metric

We use the DUC2005 and DUC2006 data sets to evaluate our proposed method empirically, where benchmark data sets are from DUC[1] for automatic summarization evaluation. DUC2005 and DUC2006 data sets consist of 50 topics. Table 2 gives a brief description of the data sets. The task is to create a summary of no more than 250 words. In those document data sets, stop words were removed using the stop list[2] and the terms were stemmed using the Porter's scheme[3] which is a commonly used algorithm for word stemming in English. For evaluation we used the Recall Oriented Understudy for Gisting Evaluation (ROUGE) evaluation package [13], which compares various summary results from several summarization methods with summaries generated by humans. ROUGE is adopted by DUC as the official evaluation metric for text summarization. It has been shown that ROUGE is very effective for measuring document summarization. It measures how well a machine summary overlaps with human summaries using N-gram co-occurrence statistics, where an N-gram is a contiguous sequence of N words. Multiple ROUGE metrics are defined according to different N and different strategies, such as ROUGE-N, ROUGE-L, ROUGE-W, ROUGE-S, and ROUGE-SU. The ROUGE-N measure compares Ngrams of two summaries, and counts the number of matches. The measure is computed by formula [13].

$$RougeN = \frac{\sum_{S\,\epsilon\,Summ_{ref}} \sum_{Ngram\,\epsilon\,S} count_{match}(Ngram)}{\sum_{S\,\epsilon\,Summ_{ref}} \sum_{Ngram\,\epsilon\,S} count(Ngram)}$$

where N stands for the length of the N-gram, Count(N-gram) is the number of N-grams in the reference summaries, and the maximum number of N-grams co-occurring in candidate summary and the set of reference summaries is $Count_{match}Ngram$.

[1] http://www.duc.nist.gov.

[2] ftp://ftp.cs.cornell.edu/pub/smart/english.stop.

[3] http://www.tartarus.org/martin/PorterStemmer/.

4.2 Comparison with Related Methods

We compare mFLSum with the most relevant methods to examine the effectiveness of the method for summarization performance improvement. These summarization methods are selected as the most efficient and the most widely used optimization summarization methods. Although there are, for each year, more than 30 systems that have participated in DUC competition, here we only compare with the DUC human best, the DUC human average, the DUC system best and the DUC system average result.

Tables 1 and 2 show the ROUGE scores of different methods using DUC2005 and DUC2006 data sets, respectively. The higher ROUGE score indicates the better summarization performance. The number in parentheses in each table slot shows the ranking of each method on a specific data set. Even thought our results are not among the best we show that sacrificing a rather small percentage of recall and precision in the quality of the produced summary can provide basis for using simple and straightforward optimization method in query focused document summarization.

Table 1. Evaluation of the methods on the DUC2005 dataset.

Summarizers	Rouge-1	Rouge-2	Rouge-SU4
S15	–	0.0725(2)	0.1316(3)
S17	–	0.0717(3)	0.1297(4)
S10	–	0.0698(4)	0.1253(5)
Centroid	0.3535(2)	0.0638(5)	0.1330(2)
CoreTerm	0.3998(1)	0.0890(1)	0.1433(1)
mFLSum	0.3350(3)	0.5173(6)	0.1144(6)

Table 2. Evaluation of the methods on the DUC2006 dataset.

Summarizers	Rouge-1	Rouge-2	Rouge-SU4
S24	–	0.0956(1)	0.1553(1)
S15	–	0.0910(2)	0.1473(3)
S12	–	0.0898(3)	0.1476(2)
Centroid	0.3807(3)	0.0785(6)	0.1330(6)
CoreTerm	0.3998(1)	0.0890(4)	0.1443(5)
mFLSum	0.3987(2)	0.0837(5)	0.1461(4)

5 Conclusion and Future Work

The paper has formalized the problem of the query-focused document summarization as the multi facility location problem. Additionally, paper has presented our study of how to incorporate query information in the own nature of mFLP. The paper has found that mFLSum is an effective summarization method. Experimental results on the DUC2005 and DUC2006 datasets demonstrate the effectiveness of the proposed approach, which compares well to most of the existing optimization methods in the literature. In future the performance of mFLSum may be further improved.

There are many potential directions for improvements such as: (1) employing sophisticated methods for the query processing/expansion techniques; (2) mFLSum currently only uses a simple syntactic and statistical features for sentence similarity calculation, in future work WordNet could be used to calculate the semantic similarity between sentences by using the synonyms sets of their component terms; (4) another possible enhancement can be reached by introducing the multi-layered graph model that emphasizes not only the sentence to sentence relations but also the influence of the under sentence and above term level relations, such as n-grams, phrases and semantic role arguments levels [14, 15].

References

1. Canhasi, E., Kononenko, I.: Multi-document summarization via archetypal analysis of the content-graph joint model. Knowl. Inf. Syst. **41**(3), 821–842 (2014)
2. Canhasi, E., Kononenko, I.: Weighted hierarchical archetypal analysis for multi-document summarization. Comput. Speech Lang. **37**, 24–46 (2016)
3. Canhasi, E.: Fast document summarization using locality sensitive hashing and memory access efficient node ranking. Int. J. Electr. Comput. Eng. **6**(3), 945 (2016)
4. McDonald, R.: A Study of Global Inference Algorithms in Multi-document Summarization. Springer, Heidelberg (2007)
5. Filatova, E., Hatzivassiloglou, V.: A formal model for information selection in multi-sentence text extraction. In: Proceedings of the 20th COLING, pp. 397–403 (2004)
6. Yih, W., Joshua, G., Vanderwende, L., Suzuki, H.: Multi-document summarization by maximizing informative content-words. In: Proceedings of IJCAI, pp. 1776–1782 (2007)
7. Takamura, H., Okumura M.: Text summarization model based on maximum coverage problem and its variant. In: Proceedings of the 12th EACL, pp. 781–789 (2009)
8. Huang, L., et al.: Modeling document summarization as multi-objective optimization. In: 2010 Third International Symposium on Intelligent Information Technology and Security Informatics (IITSI). IEEE (2010)
9. Alguliev, R.M., Aliguliyev, R.M., Isazade, N.R.: Formulation of document summarization as a 0–1 nonlinear programming problem. Comput. Ind. Eng. **64**(1), 94–102 (2013)
10. Drezner, Z., Hamacher, H.W. (eds.): Facility Location: Applications and Theory. Springer, Heidelberg (2004)
11. Carbonell, J., Goldstein, J.: The use of MMR, diversity-based reranking for reordering documents and producing summaries. In: Proceedings of ACM SIGIR, pp. 335–336 (1998)
12. Salton, G., Wong, A., Anita, A., Chung-S, Y.: A vector space model for automatic indexing. Commun. ACM **18**(11), 613–620 (1975)
13. Lin, C.-Y., Hovey, E.: Automatic evaluation of summaries using N-gram cooccurence statistics. In: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, HLT-NAACL, p. 7178 (2003)
14. Yan, S., Xiaojun, W.: SRRank: leveraging semantic roles for extractive multi-document summarization. IEEE/ACM Trans. Audio Speech Lang. Process. **22**(12), 2048–2058 (2014)
15. Canhasi, E., Kononenko, I.: Weighted archetypal analysis of the multi-element graph for query-focused multi-document summarization. Expert Syst. Appl. **41**(2), 535–543 (2014)

Query Focused Multi-document Summarization Based on Five-Layered Graph and Universal Paraphrastic Embeddings

Ercan Canhasi(✉)

Gjirafa, Inc., Rr. Rexhep Mala, 28A, Prishtine, Kosovo
ercan@gjirafa.com
http://www.gjirafa.com

Abstract. Query focused multi-document summarization is a process of automatic query biased text compression of a document set. Lately, the graph-based and ranking methods have been intensively attracted the researchers from extractive document summarization domain. The uniform sentence connecteness or non-uniform document-sentence connecteness, such as sentence similarity weighted by document importance, were the main features used by work to date. Contrary, in this paper we present a novel five-layered heterogeneous graph model. It emphasizes not only sentence and document level relations but also the influence of lower level relations (e.g. a part of sentence similarity) and higher level relations (i.e. query to sentences similarity). Based on this model, we developed an iterative sentence ranking algorithm, based on the existing well known PageRank algorithm. Moreover, for text similarity calculations we used universal paraphrase embeddings that outperform various strong baselines on many text similarity tasks and many domains. Experiments are conducted on the DUC 2005 data sets and the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) evaluation results demonstrate the advantages of the proposed approach.

Keywords: Multidocument summarization · Graph-based summarization · Graph-based ranking algorithm · PageRank

1 Introduction

Query oriented multi-document summarization (QMDS) [1,2] targets to select the most central content from a group of documents to produce a compact query focused summary. Lately, the graph-based and ranking methods have been intensively attracted the researchers from extractive document summarization domain.

The uniform sentence connecteness or non-uniform document-sentence connecteness, such as sentence similarity weighted by document importance, were the main features used by research to date. Contrary, in this paper we present a novel five-layered heterogeneous graph model. It emphasizes not only sentence

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_22

and document level relations but also the influence of lower level relations (e.g. a part of sentence similarity) and higher level relations (i.e. query to sentences similarity). We denote the given group of documents D=d_1,d_2,...d_n as a weighted undirected frame graph G. Alternatively to earlier work, our graphs cover not only one type of objects (i.e. sentences), instead they cover five sort of objects: semantic role frames, sentences, paragraphs, documents and query.

Despite the fact that human annotators usually disagree on the content of the gold summary, they perform very well on this task. Thus, our primary target in this work was to conceptualize the summarizer based on the cognition behind the human equivalent summarizer. In doing so, we examine employing the psychology cognitive situation pattern, namely the Event-Indexing model [3]. Whilst reading the document a human equivalent system, based on this patter, needs to evidence differences on five dimensions. Which are, given in descending order of importance: *(1) protagonist*, *(2) temporality*, *(3) spatiality*, *(4) causality* and *(5) intention*. Fortunately, the semantic role parser's [9] output can be formulated as the above proposed cognitive pattern. The connection between syntactic constituents and the predicates make the core of the frames known as Semantic roles. Most sentence components have semantic connections with the predicate, carrying answers to the questions such as who, what, when, where etc.

Mostly, the text similarity evaluation is done using the standard vector space model. In this work, in order to employ some novel methods for text representation, we used universal paraphrase embeddings [4] for text representation and similarity calculation. This representation outperform various strong baselines on many text similarity tasks and many domains. We are not aware of any other paraphrase models that perform nearly as well out-of-the-box as this method. Moreover, this representations are very easy to use. They do not require the use of any neural network architecture, instead the embeddings can be downloaded and then summed for a given phrase/sentence inside of an application to create the phrase/sentence embedding. Moreover, we also find that this representations can improve general text similarity models.

The summarization technique presented in this work operates as follows. Initially, the SRL parser is used to extract semantic arguments from each treated sentence. Next, the composite similarity among all semantic frames is calculated using the event-indexing pattern. Then a semantic graph is produced. Produced graph consist of nodes, which are semantic frames, and arcs, which are the composite similarity values. Next, utilizing an inventive weighting strategy four more layers, namely the sentence, paragraph, document and query layers. Adding new layers allows producing richer multi-layered graph pattern with the inter and intra sentence, paragraph, document and query stage affinities. Later, adjusted form of PageRank is utilized in order to spot the important nodes in the graph. Next, in order to remove more repetitious information the sentences extensively overlying with other top rated sentences are penalized. Based on the text graph and the obtained rank scores, a greedy algorithm is applied to inflict the diversity penalty and compute the final rank scores of the sentences. Later, we sum the PageRank scores of semantic frames, originating from the same sentence, and we use it as a score for sentence scoring. Finally, we extract the most significant sentences, starting from the sentence with highest rank.

The rest of the paper is arranged as pursues. In Sect. 2 the survey of existing graph-based summarization models is presented. Sections 3 and 4 presents the projected sentence ranking algorithm. Next, in Sect. 5 the experiments and evaluations are given. Finally, Sect. 6 offers conclusion and some direction for future work.

2 Related Work

Lately, the extractive document summarization community has been extensively researching the graph based models [5,6]. A document or a set of documents are generaly trated as a text similarity graph formed by denoting a text units as a nodes and similarity between text units as arcs. Weighting the relevance of a node in a graph is usualy done via a graph-based ranking algorithms, such as PageRank [7] or HITS [8]. The common final step in a graph based summarization methodology is sentence extraction. In it, the sentences from document(s) are rated based on their node importance and the top ranked ones are putten into final extractive summary. The pioneering algorithm called LexRank [5], adapted from PageRank, the most typically used in summarization, is applied to calculate sentence significance. Similarely, the work of Mihalcea and Tarau [6], yet another PageRank variation called TextRank, can be used for same purpose.

3 Five-Layered Graph Model

This section presents a graph model, which in return will be used in a frame ranking algorithm presented in next section. Let us represent the set of documents C as a graph $\varGamma = (N_f, N_s, N_p, N_d, N_q, A^{N_f}, A^{N_s}, A^{N_p}, A^{N_d}, A^{N_q}, \alpha_n, \beta_n, \gamma_n, \delta_n, \epsilon_n, \alpha_a, \beta_a, \gamma_a, \delta_a, \epsilon_a)$, where N_f, N_s, N_p, N_d and N_q represent the frame, sentence, paragraph, document and query nodes set, respectively. $A^{N_f} \subseteq N_f \times N_f, A^{N_s} \subseteq N_s \times N_s$, $A^{N_d} \subseteq N_d \times N_d$, $A^{N_p} \subseteq N_p \times N_p$ and $A^{N_q} \subseteq N_q \times N_q$ are frame, sentence, paragraph, document and query arc set. $\alpha_n : N_f \rightarrow \Re_+, \beta_n : N_s \rightarrow \Re_+$, $\gamma_n : N_p \rightarrow \Re_+$, $\delta_n : N_d \rightarrow \Re_+$ and $\epsilon_n : N_q \rightarrow \Re_+$ are five functions defined to label frame, sentence, paragraph, document and query vertices, while $\alpha_a : A^{N_f} \rightarrow \Re_+, \beta_a : A^{N_s} \rightarrow \Re_+$, $\gamma_a : A^{N_p} \rightarrow \Re_+$, $\delta_a : A^{N_d} \rightarrow \Re_+$ and $\epsilon_a : A^{N_q} \rightarrow \Re_+$ are functions for labeling frame, sentence, paragraph, document and query arcs.

In order to further augment the actual PageRank algorithm five additional layers are joined to the original sentence similarity graph which in return produces the superior five-layered graph model with the inter and intra sentence, paragraph and document level relations. In Fig. 1 the previous typical text graph model and one after applying the concepts of five-layered graph is shown. As shown by Fig. 1 the new graph model reveals some formerly neglected information such as: (1) sentence to sentence similarity can now be differentianted in four groups, one within a document, one across two documents, one within a

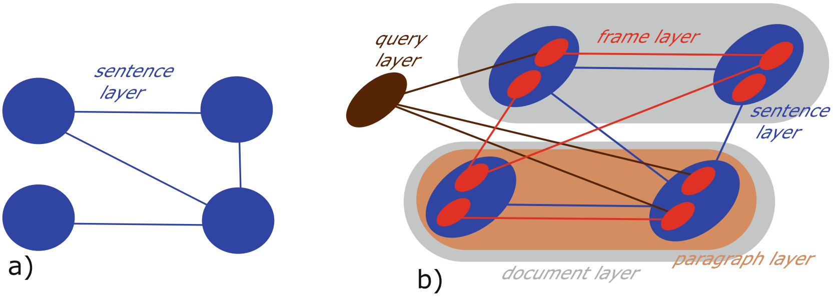


Fig. 1. Summarization graph model (a) before and (b) after introducing multilayered model.

paragraph, another across two paragraphs, (2) the documents and paragraphs magnitudes can influence the sentence rankings; (3) there is also a entirely original kind of objects (i.e. semantic role labeler (SRL) [9] frames) participating in defining the inner sentence connections. Since our main aim is query oriented summarization, we additionally heavily exploit the idea of query to sentences similarity which is in return directly integrated in very nature of our model.

The sentence arc function $\beta_a(s_i, s_j) = sim_{normal}(s_i, s_j) = \frac{sim(s_i,s_j)}{\sum_{s_k \in S \wedge k \neq i} sim(s_i,s_k)}$, paragraph arc function $\gamma_a(p_i, p_j) = sim_{normal}(p_i, p_j) = \frac{sim(p_i,p_j)}{\sum_{p_k \in D \wedge k \neq i} sim(p_i,p_k)}$, document arc function $\delta_a(d_i, d_j) = sim_{normal}(d_i, d_j) = \frac{sim(d_i,d_j)}{\sum_{d_k \in D \wedge k \neq i} sim(d_i,d_k)}$ and query arc function $\epsilon_a(s_i, q) = sim_{normal}(s_i, q) = \frac{sim(s_i,q)}{\sum_{s_k \in S \wedge k \neq i} sim(s_i,q)}$ are defined as the normalized similarity between a two sentences s_i and s_j, the normalized similarity between a two paragraphs p_i and p_j, two documents d_i and d_j, and between a sentence. The SRL frame arc function $\alpha_a(f_i, f_j) = sim_{composite}(f_i, f_j)$, is draft as the conglomerated similarity function of two frames f_i and f_j. Suppose F is the overall count of frames in a documents set. The frame nodes function $\alpha_n(f_i)$ = nominates to frame vertices the value of $1/F$ or $1/2F$, counting on completeness, where incomplete frames have lower weight. The sentence $\beta_n(s_i) = centr\_norm(s_i) = \frac{\sum_{u \in s_i} cw(u)}{\sum_{v \in S} cw(v)}$ and document nodes $\gamma_n(d_i) = centr\_norm(d_i) = \frac{\sum_{u \in d_i} cw(u)}{\sum_{v \in D} cw(v)}$ functions are represented with the normalized centroid-based weight of the sentence and document, respectively where $cw(u)$ denotes the centroid [10] weight of word u.

Since our primary objective is to apprehend the similarity and redundancy between sentences, especially at a subordinate structural and a surpassing semantic degree, we used the event-indexing model as the base for calculations of semantic similarity between frames of semantic role parser outputs, namely frames. Denote the similarity measure for protagonist as $sim_{protagonist}(f_i, f_j) = \alpha_1 \cdot sim(A0_i, A0_j) + \alpha_2 \cdot sim(A1_i, A1_j) + \alpha_3 \cdot sim(A2_i, A2_j) + \alpha_4 \cdot sim(A0_i, A1_j) + \alpha_5 \cdot sim(A0_i, A2_j) + \alpha_6 \cdot sim(A1_i, A2_j)$; temporality as $sim_{temporality}(f_i, f_j) = sim(Am\_Tmp_i, Am\_Tmp_j)$; spatiality as

$sim_{spatiality}(f_i, f_j) = sim(Am\_Loc_i, Am\_Loc_j)$ and causality as

$$sim_{causality}(f_i, f_j) = sim(Predicate_i, Predicate_j).$$

In pursuance of the more adaptive weighting strategy we use coefficients $\alpha_1 = \alpha_2 = \alpha_3 = 0.25; \alpha_4 = \alpha_5 = 0.10; \alpha_6 = 0.5$. Consequently, the compose similarity can be assign as:

$$\begin{aligned} sim_{composite}(f_i, f_j) = &\Big(\beta_1 sim_{protagonist}(f_i, f_j) + \beta_2 sim_{temporality}(f_i, f_j) \\ &+ \beta_3 sim_{spatiality}(f_i, f_j) + \beta_4 sim_{causality}(f_i, f_j)\Big)/\#arguments \end{aligned}$$

By reason of the cognitive model which permits the weights of different dimensions in decreasing order of importance we set the values for coefficients as $\beta_1 = 0.4; \beta_2 = 0.3; \beta_3 = 0.2; \beta_4 = 0.1$.

4 Five-Layered Graph-Based Ranking Algorithm

Given the solution for multilayered text similarity graph, presented in previous section, in current section a adjusted iterative graph-based sentence ranking algorithm is proposed.

Our algorithm is continued from those existing PageRank-like algorithms described in the literature. Differently from our methods they used to calculate the graph only in the sentence level [5,6], or sentence and document level [11–13].

In essence, PageRank method (in matrix notation) as reported in the original paper [7] is

$$\pi^{(k+1)T} = \alpha\pi^{(k)T}\mathbf{H} + (\alpha\pi^{(k)T}a + 1 - \alpha)\mathbf{v}^T$$

where $\mathbf{H}$ is a very sparse, raw sub stochastic hyperlink matrix, α is a scaling parameter between 0 and 1, π^T is the stationary row vector of $\mathbf{H}$ called the PageRank vector, $\mathbf{v}^T$ is a complete dense, rank-one teleportation matrix and a is a binary dangling node vector. In the sentence ranking terminology the matrix $\mathbf{H}$ is an adjacency matrix of similarities, $\mathbf{v}^T$ is the affinity vector and the resulting π^T is the frame ranking vector.

For purpose of simplicity, we consider there are two documents (e.g. D_1, D_2) and 4 sentences (e.g. $S_{1,1}, S_{1,2}$ in D_1 and $S_{2,1}, S_{2,2}$ in D_2) involved in ranking. Yet should be eight frames extracted from four sentences, let us show the document, sentence (just the first one) and frames (again just the first one) similarity matrices:

$$H_0 = \begin{bmatrix} D_{1,1} & D_{1,2} \\ D_{2,1} & D_{2,2} \end{bmatrix} D_{1,1} = \begin{bmatrix} P_{1,1} & P_{1,2} \\ P_{2,1} & P_{2,2} \end{bmatrix} P_{1,1} = \begin{bmatrix} S_{1,1} & S_{1,2} \\ S_{2,1} & S_{2,2} \end{bmatrix} S_{1,1} = \begin{bmatrix} F_{1,1} & F_{1,2} \\ F_{2,1} & F_{2,2} \end{bmatrix}$$

$$H_{DQ_{1,1}} = \begin{bmatrix} w_{SQ_{11}} S_{1,1} & 0 \\ 0 & w_{SQ_{22}} S_{2,2} \end{bmatrix} H_{D_{1,1}} = \begin{bmatrix} w_{P_{11}} P_{1,1} & w_{P_{12}} P_{1,2} \\ w_{P_{21}} P_{2,1} & w_{P_{22}} P_{2,2} \end{bmatrix}$$

$$H_{P_{1,1}} = \begin{bmatrix} w_{S_{11}} S_{1,1} & w_{S_{12}} S_{1,2} \\ w_{S_{21}} S_{2,1} & w_{S_{22}} S_{2,2} \end{bmatrix} H_{S_{1,1}} = \begin{bmatrix} w_{F_{11}} F_{1,1} & w_{F_{12}} F_{1,2} \\ w_{F_{21}} F_{2,1} & w_{F_{22}} F_{2,2} \end{bmatrix}$$

The block matrix $D_{1,1}$ attributes to the similarity matrix of paragraphs in document D_1, while $D_{1,2}$ denotes the fold-document (D_1 and D_2) similarity matrix, and so on and so forth. The block matrix $P_{1,1}$ attributes to the similarity matrix of sentences in paragraph P_1, while $P_{1,2}$ represents the fold-document (P_1 and P_2) similarity matrix, and so on and so forth. Similarly the block matrix $S_{1,1}$ in $D_{1,1}$ represents the similarity matrix of the frames in sentence S_1, while $S_{1,2}$ denotes the fold-sentence (S_1 and S_2) affinity matrix. H_0 correlates to the original sentence similarity matrix used in generic graph methods. In order to advertise the document and sentence weight on the sentence and frame arcs that connect different documents and sentences, the most sufficient way is to integrate the document and sentence dimension into the H_0 as illustrated in formulas for $H_{D_{1,1}}$ and $H_{S_{1,1}}$. The weight matrix $W_{sq} = \begin{bmatrix} w_{SQ_{11}} & 0 \\ 0 & w_{SQ_{22}} \end{bmatrix}$ is needed to weight each sentence based on sentence-query similarity, the weight matrix $W_d = \begin{bmatrix} w_{P_{11}} & w_{P_{12}} \\ w_{P_{21}} & w_{P_{22}} \end{bmatrix}$ is handled to discriminate the cross-document paragraph arcs, the weight matrix $W_p = \begin{bmatrix} w_{S_{11}} & w_{S_{12}} \\ w_{S_{21}} & w_{S_{22}} \end{bmatrix}$ is on the other side employed to differentiate the cross-document sentence arcs and the weight matrix, likewise $W_s = \begin{bmatrix} w_{F_{11}} & w_{F_{12}} \\ w_{F_{21}} & w_{F_{22}} \end{bmatrix}$ is used to distinguish the cross-sentence frame arcs. The diagonal elements in W_{sq} are calculated as sentence to query similarity, while non diagonal elements are set to 0. The diagonal elements in W_d, W_p, and W_s are set to 1 to neutralize the influence of the intra-document sentence and intra-sentence frame arcs. Just on the opposite, the non-diagonal elements are weighted by the connections between the two corresponding documents and sentences. We define $W_{sq}(i) = 1 + \delta_a(s_i), q)$. We define W_p as $W_p(i,j) = 1 + \gamma_a(p(s_i), p(s_j))$, where $p(s_i)$ presents the document that contains the sentence s_i. We define W_d as $W_d(i,j) = 1 + \delta_a(d(p_i), d([_j))$, where $d(p_i)$ presents the document that contains the paragraph p_i. We also define W_s as $W_s(i,j) = 1 + \beta_a(s(f_i), s(f_j))$, and $s(f_i)$ presents the sentence that contains the frame f_i.

Given that a frame originating from the document, paragraph and the sentence with higher significance should be ranked even higher, we echo their importance via the affinity vector $\boldsymbol{v}$. Therefore, the centroid-based weight of latter are taken as the weights on the affinity vector $\boldsymbol{v}$. See the following preference vectors:

$$\boldsymbol{v}_o = [\boldsymbol{v}_{d_1}\boldsymbol{v}_{d_2}]^T; \boldsymbol{v} = \left([\boldsymbol{v}_{d_1}\boldsymbol{v}_{d_2}] \cdot \begin{bmatrix} w_{d_1} & \\ & w_{d_2} \end{bmatrix}\right)^T; \boldsymbol{v}_{d_1} = \left([\boldsymbol{v}_{p_1}\boldsymbol{v}_{p_2}] \cdot \begin{bmatrix} w_{p_1} & \\ & w_{p_2} \end{bmatrix}\right)^T;$$

$$\boldsymbol{v}_{p_1} = \left([\boldsymbol{v}_{s_1}\boldsymbol{v}_{s_2}] \cdot \begin{bmatrix} w_{s_1} & \\ & w_{s_2} \end{bmatrix}\right)^T; \boldsymbol{v}_{q_1} = \left([\boldsymbol{v}_{sq_1}\boldsymbol{v}_{sq_2}] \cdot \begin{bmatrix} w_{sq_1} & \\ & w_{sq_2} \end{bmatrix}\right)^T;$$

Here $\boldsymbol{v}_o$ represents the original preference vector, as used in LexRank, $\boldsymbol{v}_{s_1}$ the sub-preference vector of the frames from sentence s_1; $\boldsymbol{v}_{d_1}$ denotes the sub-preference vector of the paragraphs from document d_1, and $\boldsymbol{v}_{p_1}$ denotes the sub-preference vector of the sentences from paragraph p_1.

Lemma 1. *$\boldsymbol{v}$ is a probability vector, if W_{v_d}, W_{v_p}, W_{v_q}, W_{v_s} are positive and the diagonal elements in them sum to 1;*

Proof. Since $\boldsymbol{v}_{s_1}, ... \boldsymbol{v}_{s_n}$ are probability vectors of frames from sentences we have $|\boldsymbol{v}_{s_1}| = ... = |\boldsymbol{v}_{s_n}| = 1$. Then,

$$\begin{aligned}|\boldsymbol{v}| &= w_{d_1}(w_{pd_1}(w_{sp_1}(w_{sq_1}|\boldsymbol{v}_{s_1}| + ... + w_{sq_n}|\boldsymbol{v}_{sp_n}|)\\ &\quad + ... + w_{d_n}(w_{pd_n}(w_{sp_n}(w_{sq_1}|\boldsymbol{v}_{sq_1}| + ... + w_{sq_n}|\boldsymbol{v}_{sq_n}|)\\ &= w_{d_1}(w_{pd_1}(w_{sp_1}(w_{sq_1} + ... + w_{sq_n}) + ... + w_{d_n}(w_{pd_n}(w_{sp_n}(w_{sq_1} + w_{sq_n}) = 1\\ &\quad \text{if } (w_{sq_1} + ... + w_{sq_n}) = 1 \text{ hence } w_{d_1}w_{pd_1}w_{sp_1} + ... + w_{d_n}w_{sp_n}w_{pd_n} = 1\end{aligned}$$

Next, the matrix H should be transformed in column stochastic and irreducible through forcing each of four block matrices of sentences and two matrices of documents to be column stochastic simply by normalizing them by columns. The sixteen block matrices in H, therefore H itself, should also be made irreducible via adding additional links between any two frames, which is also adapted in PageRank.

Lemma 2. *H is column stochastic and irreducible.*

Proof. H is column stochastic since the weight matrix W is column stochastic. Let A, B, C, and D, donate any of the 4 column block in H, then

$$\begin{aligned}\sum_i H_{ij} &= w_{d_{1k}}\left(w_{p_{1k}}\left(w_{sq_{1k}}\left(w_{s_{1k}}\sum_i A_{ij} + w_{s_{2k}}\sum_i B_{ij}\right)\right)\right)\\ &\quad + w_{d_{2k}}\left(w_{p_{2k}}\left(w_{sq_{2k}}\left(w_{s_{3k}}\sum_i C_{ij} + w_{s_{4k}}\sum_i D_{ij}\right)\right)\right)(k = 1, \ldots, 4)\\ \sum_i H_{ij} &= w_{d_{1k}}(w_{p_{1k}}(w_{sq_{1k}}(w_{s_{1k}} + w_{s_{2k}})) + w_{d_{2k}}(w_{p_{2k}}(w_{sq_{2k}}(w_{s_{3k}} + w_{s_{4k}}))\\ &\text{if } (w_{s_1} + w_{s_2}) = 1, (w_{s_3} + w_{s_4}) = 1 \text{ and } (w_{sq_1} + w_{sq_2}) = 1 \text{ hence } w_{d_1} + w_{d_2} = 1\end{aligned}$$

Since the four graphs corresponding to the four diagonal block matrices in H are strongly connected (i.e. they are irreducible) and the arcs connecting the four graphs are bidirectional, the graph corresponding to H is obviously strongly connected. Thus H must be also irreducible.

Notice that we must ensure $W_{v_{s_1}} > 0, W_{v_{s_2}} > 0$ and make the sum of the diagonal elements equal to 1 in order to ensure $\boldsymbol{v}$ to be a probability vector. And we must make H column stochastic by setting $W_d > 0, W_s > 0$ and both matrices column stochastic. Finally, we obtain that H is stochastic, irreducible and primitive, hence we can compute the unique dominant vector (with 1 as the eigenvalue) of H by using the power iteration method applied to H which converges to π. The previous explanation is given as example with a two-document, a four sentences (two of them in every document), and eight frames (two of frames in every single sentence). However, we can come to the same conclusion when the number of the documents, sentences and frames involved extends from given values to arbitrary number.

Eventually, the new ranking algorithm can be outlined as follows:

$$H(i,j) = \alpha_a(f_i, f_j)W_d(i,j)W_p(i,j)W_{sq}(i,j)W_s(i,j);$$
$$\boldsymbol{v} = \alpha_n(f_i)(1+\beta_n(s(f_i)))(1+\gamma_n(p(s_i)))(1+\delta_n(d(p_i)))(1+\epsilon(s_q)).$$

Up to this point, as it was the main purpose of this chapter, the document, paragraph, sentence and query dimensions have been joined into the PageRank-like algorithms for frame ranking based a strong mathematical basis.

5 Evaluation

In this section, we conduct experiments on a DUC [1] data set from 2005 to evaluate the effectiveness and possible positive contributions of the proposed method compared with other existing summarization systems. We use the DUC2005 data sets to evaluate our proposed method empirically, where benchmark data sets are from DUC for automatic summarization evaluation. DUC2005 data set consist of 50 topics. The task is to create a summary of no more than 250 words. In those document data sets, stop words were removed using the stop list and the terms were stemmed using the Porters scheme which is a commonly used algorithm for word stemming in English.

Table 1. ROUGE-1 scores of the DUC 2005 and evaluation of our model

Summarizers	ROUGE-1	ROUGE-2	ROUGE-SU4
Avg-Human	0.4417 (1)	0.1023 (1)	0.1622 (1)
Avg-DUC05	0.3434 (7)	0.0602 (6)	0.1148 (7)
System-15	0.3751 (4)	0.0725 (4)	0.1316 (4)
System-4	0.3748 (5)	0.0685 (5)	0.1277 (5)
Biased-Lex	0.3861 (3)	0.0753 (3)	0.1363 (2)
Our system	0.3880 (2)	0.0793 (2)	0.1350 (3)

Machine-generated summaries are evaluated using ROUGE [14] automatic n-gram matching which measures performance based on the number of co-occurrences between machine-generated and ideal summaries in different word units. The 1-gram ROUGE score (a.k.a. ROUGE-1) has been found to correlate very well with human judgements at a confidence level of 95%, based on various statistical metrics. Even though in this version of method we did not consider sentence positions or other summary quality improvement techniques such as sentence reduction, its overall performance is promising, please see Table 1. The use of multilayered model in summarization can make considerable improvements even though the results presented here report a minimal improvements.

[1] Document Understanding Conference (http://duc.nist.gov).

6 Conclusion and Future Work

The five-layered graph model and supporting methods for node ranking with their application to query focused summarization was presented in this paper. The central improvement presented in this work is the concept of 5-layered graph model. Exploiting proposed methods in query oriented summarization results in quality improvements. Nevertheless, improvements are quite possible. The main future extension of this work should include an extensive investigation on effects of each additional layer. Presented method can be advanced even more: (1) especially in taking advantage of the universal paraphrastic embeddings, and (2) the possibility of its application to update and compare summarization. We are also working on further improvements of the model, and it's adaptation to other summarization tasks, such as the update and compare summarization [15].

References

1. Canhasi, E., Kononenko, I.: Weighted archetypal analysis of the multi-element graph for query-focused multi-document summarization. Expert Syst. Appl. **41**(2), 535–543 (2014)
2. Canhasi, E.: Fast document summarization using locality sensitive hashing and memory access efficient node ranking. Int. J. Electr. Comput. Eng. **6**(3), 945 (2016)
3. Zwaan, R.A., Langston, M.C., Graesser, A.C.: The construction of situation models in narrative comprehension: an event-indexing model. Psychol. Sci. **6**(5), 292–297 (1995)
4. Wieting, J., Bansal, M., Gimpel, K., Livescu, K.: Towards universal paraphrastic sentence embeddings. arXiv preprint arXiv:1511.08198 (2015)
5. Erkan, G., Radev, D.R.: Lexrank: graph-based lexical centrality as salience in text summarization. J. Artif. Intell. Res. (JAIR) **22**, 457–479 (2004)
6. Mihalcea, R., Tarau, P.: Textrank: bringing order into text. In: EMNLP, pp. 404–411 (2004)
7. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. Comput. Netw. **30**(1–7), 107–117 (1998)
8. Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. J. ACM **46**(5), 604–632 (1999)
9. Carreras, X., Marque, L.: Introduction to the conll-2004 shared task: semantic role labeling. In: CoNLL, pp. 89–97 (2004)
10. Radev, D.R., Jing, H., Sty, M., Tam, D.: Centroid-based summarization of multiple documents. Inf. Process. Manage. **40**(6), 919–938 (2004)
11. Otterbacher, J., Erkan, G., Radev, D.R.: Biased lexrank: passage retrieval using random walks with question-based priors. Inf. Process. Manage. **45**(1), 42–54 (2009)
12. Wei, F., Li, W., Qin, L., He, Y.: A document-sensitive graph model for multi-document summarization. Knowl. Inf. Syst. **22**(2), 245–259 (2010)
13. Wan, X.: Document-based HITS model for multi-document summarization. In: Ho, T.-B., Zhou, Z.-H. (eds.) PRICAI 2008. LNCS (LNAI), vol. 5351, pp. 454–465. Springer, Heidelberg (2008). doi:10.1007/978-3-540-89197-0_42
14. Lin, C.-Y., Hovy, E.H.: Automatic evaluation of summaries using n-gram co-occurrence statistics. In: HLT-NAACL (2003)
15. Canhasi, E., Kononenko, I.: Weighted hierarchical archetypal analysis for multi-document summarization. Comput. Speech Lang. **37**, 24–46 (2016)

Prediction of Diseases Using Hadoop in Big Data - A Modified Approach

D. Jayalatchumy[1(✉)] and P. Thambidurai[2]

[1] Department of CSE, PKIET, Karaikal, India
djlatchumy@gmail.com
[2] PKIET, Karaikal, India
pdurai158@gmail.com

Abstract. Big data plays an important role in healthcare. With an increase in growth of data, processing and analyzing them becomes a challenging task. Diagnosis and prediction of disease becomes difficult especially when it comes to Big Data. Clustering is one of the Data Mining tools that help us to analyze Big Data effectively. Existing algorithms have high computational complexity and they do not perform well especially when it comes to Big Data. Since most of the data is unstructured a graph based spectral technique using power method is chosen for analysis. The algorithm is made more effective by making them to converge using extrapolation technique. Moreover, they are designed to handle larger datasets using MapReduce framework. The main objective of this paper is to help the doctors in predicting the diseases more accurately using the proposed algorithm. Experiments were conducted on various synthetic datasets and real data's to prove the algorithmic efficiency and accuracy.

Keywords: Clustering · Big data · Hadoop · Extrapolation · Aitken

1 Introduction

As the size of data keeps on increasing, analyzing & processing becomes difficult. Existing tools or techniques has to be modified to withstand this growth. Clustering is one of the Big Data tool to analyze data efficiently. Spectral Clustering (SC) is an efficient clustering algorithm performed using eigen value decomposition. But, the time and space complexity of SC is very high [2]. So, Lin and Cohen [1] addressed this problem in their paper by introducing the Power Iteration Clustering algorithm that replaces eigen value decomposition by matrix vector multiplication. p-PIC designed by Weizhong et al. [5] explores different parallelization techniques for reducing communicational and computational cost. The limitations are its convergence factor, node failure and fault tolerance. They are stuck at global minima. Hence, this paper concentrates on convergence factor and its implementation on distribute framework Hadoop. Importance has been given to reduce the number of iterations that decreases the computational time. The algorithm has been tested on various synthetic and real datasets to check its effectiveness on larger data. The results show faster convergence of the algorithm and reduction in the error rate. A study has been carried out by applying the proposed algorithm on healthcare dataset collected from the hospital in the nearby surroundings for prediction of the diseases.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_23

2 Proposed Acceleration Power Method for Big Data

PIC is an algorithm that clusters data using power method. Power method finds the largest of eigen vector which is a combination of the eigen vectors in a linear manner [3, 5]. The algorithm uses the matrix vector multiplication where the matrix W is combined with the vector v_0 to obtain Wv_0. PIC is an iterative process in which the vector gets updated and normalized to avoid it becoming too large using $\frac{v_t}{||v_t||}$. The largest eigen vector using power method is [16]. $v^{t+1} = cWv^{t-1}$ where $c = \frac{1}{||Wv^t||}$.

```
Begin
    Read the dataset
    Construct the similarity matrix A∊R^{n*n}
    Obtain a row normalized affinity matrix W, Pick initial vector v_o
    Repeat
      $v^t = \gamma W v^{t-1}$ & $\delta^{t+1}$ =, where $\gamma = \frac{1}{||Wv^{t+1}||}$
      Increment t
   Until$|\delta$^$t - \delta$^$(t-1)\ | = 0$
      Cluster point on v^t using K-means and output the clusters.
  End
```

Algorithm 1. Power Iteration Clustering

The procedure for PIC is shown in Algorithm 1. Power Iteration Clustering [1] algorithm works well for larger datasets. The computational complexity is given as O (n^2) for SC and O(n) for PIC. Vector <v^t> converges to local minima that forms the cluster of the datasets. It converges more quickly than spectral method because, it stops when <v^t> stop accelerating [17]. Convergence acceleration techniques [4] are methods for reducing the computational time needed to solve a problem while preserving the accuracy of the solution. Extrapolation is the method to improve the speedup that helps in the convergence of an algorithm. It is used to accelerate the linear convergent sequence into quadratic sequence [9, 10]. Since the efficiency of an algorithm depends upon its convergence, an extrapolation technique is applied to PIC.

2.1 Aitken Method

Aitken method is an acceleration technique that increases the rate of convergence of an error sequence. The sequence converges more rapidly with less number of iterations. Let $\{p_n\}$ be a sequence which converges to its limit ‘p’ linearly [4, 10]. Then the Aitken method is performed by choosing the initial value [11]. Choose an integral guess x_0,

Step 1: Calculate p_0, p_1, p_2 using any linear iteration method.

Step 2: Compute $\widehat{p}_0 = p_0 - \frac{(p_1 - p_0)^2}{p_2 - 2p_1 + p_0}$ and for n = 1, 2……

Step 3: Compute p_{n+2}

Step 4: Compute $\widehat{p}_n = p_n - \frac{(p_{n+1} - p_n)^2}{p_{n+2} - 2p_{n+1} + p_n}$
Step 5: Terminate if $|\widehat{p}_n - \widehat{p}_{n-1}| < \in \& p \approx \widehat{p}_{n.}$

2.2 Error Estimation

For an iteration to converge, the error must decrease between each successive value of x_n. The error for each iteration is given as [6, 8]

$$e_n = x_n - x^* \ldots\ldots\ldots\ldots\ldots \quad (1)$$

x* is the fixed point at which the sequence will converge $g(x^*) = x^*$; $g(x_n) = x_{n+1}$ Therefore from (1),

$$e_{n+1} = g(x_n) - g(x^*) \ldots\ldots\ldots \quad (2)$$

From Taylors theorem on approximation $g(x_n)$ at x^*

$$g(x_n) \approx g(x^*) + g'(x^*) g(x^* - x_n) = g(x^*) + g'(x^*)$$

Sub it in (2), we get

$$e_{n+1} \approx g(x^*) + g'(x^*) e_n - g(x^*) \quad (i.e) e_{n+1} \approx g'(x^*) e_n \ldots\ldots \quad (3)$$

The error rate is shown in Eq. (2). For the error rate to decrease and the series to converge $g'(x^*)e_n$ must be less than 1. The error between the iteration has been calculated using the Aitken acceleration method as shown in Eq. (3) and it's found to be decreased by 9%.

3 Aitken Power Iteration Clustering Algorithm

The rate of convergence of power method is linear. The convergence sequence of constants is also linear. Hence, Aitken Δ^2 method can be used as it is a linear convergent method to form a new sequence that can converge faster. Aitken Δ^2 can be adapted to speed up the convergence of power method since it computes the principle eigen vector of a matrix in one step. The fixed point iteration with acceleration by Aitken's Δ^2 method generates the sequence $\{\widehat{x}_k\}$ where [7, 11]

$$\widehat{x}_k = x_k - \frac{(\Delta x_k)^2}{\Delta^2 x_k}$$

The embedding of Aitken extrapolation in power method is shown in Algorithm 2.

```
function $\vec{x}^{n}$= Aitken Power Method(){
    $\vec{x}^{(c)}$=$\vec{v}$;  $k = 1$;
    repeat
        $\vec{x}^{(k)}$=$\vec{A}_x^{(k-1)}$ ;
        $\delta$=||$x^{(k+1)} - x^{(k)}$||;
        $\vec{x}^{(k)}$=Aitken($\vec{x}^{(k+c)}$,$\vec{x}^{(k+1)}$,$\vec{x}^{(k)}$);
        $k = k + 1$;
    until $\delta$<c;    }
```

Algorithm 2. Aitken Power Iteration Clustering

4 MapReduce Architecture

MapReduce is a programming model and software framework developed by Google. It is a method for large scale parallelization. It processes large amount of data in a reliable, fault tolerant manner. It does computation on both structured and unstructured data. It has 2 components, HDFS and MapReduce [15].

HDFS hold a large amount of data across multiple machines. It is suitable for distributed storage. The name node and data node helps to check the status of the cluster [13]. Name node manages the file system namespace and access clients file. It performs operation like open, close etc. Data node performs operations based on the request. The block creation, deletion, replication operation are performed by the data node. The Map takes the dataset and converts it into the <key value> pairs. The map task is done by the class that takes the input and sorts them [12, 13]. The output of the mapped class is used as input by reducer class. The partitioner partitions the key value pair of the mapped output. The reduce task gets the output from map and combine them into smaller set of tuples. The architecture of MapReduce is shown in Fig. 1.

Cluster Setup. The Hadoop cluster has 3 components client, master and the slave. The client loads the data into the cluster, submits the job and describes how the data should be processed [14]. It retrieves data after job completion. The slave stores the data and processes them. The cluster set up is shown in Fig. 2. The procedure for Aitken power method in MapReduce is shown in Algorithm 2. The Hadoop cluster supports 3 modes. Local (standalone), Pseudo distributed mode and Multi or fully distributed mode.

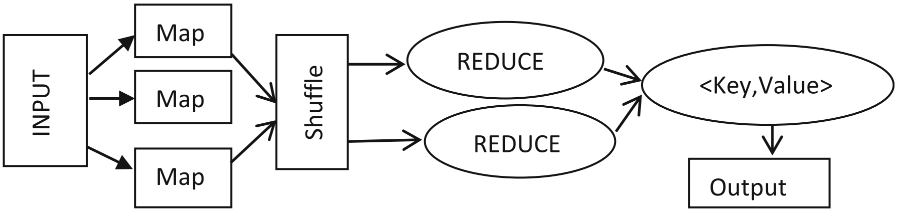
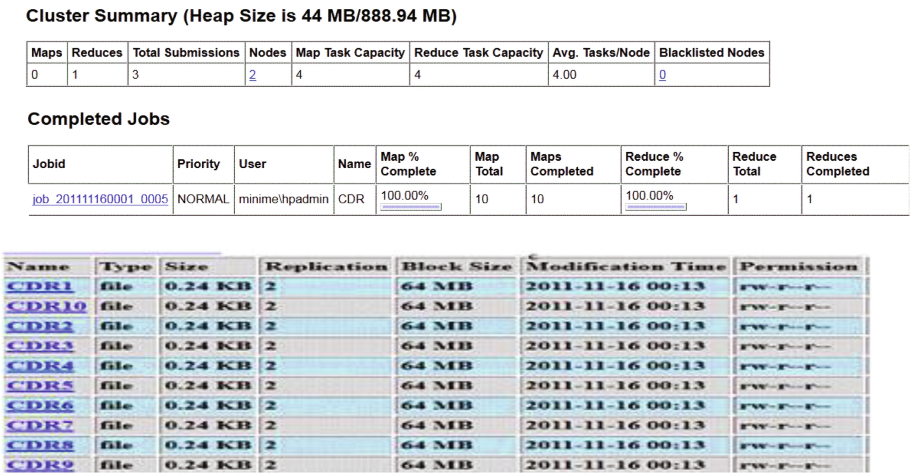


Fig. 1. MapReduce architecture



Cluster Summary (Heap Size is 44 MB/888.94 MB)

Maps	Reduces	Total Submissions	Nodes	Map Task Capacity	Reduce Task Capacity	Avg. Tasks/Node	Blacklisted Nodes
0	1	3	2	4	4	4.00	0

Completed Jobs

Jobid	Priority	User	Name	Map % Complete	Map Total	Maps Completed	Reduce % Complete	Reduce Total	Reduces Completed
job_201111160001_0005	NORMAL	minime\hpadmin	CDR	100.00%	10	10	100.00%	1	1

Name	Type	Size	Replication	Block Size	Modification Time	Permission
CDR1	file	0.24 KB	2	64 MB	2011-11-16 00:13	rw-r--r--
CDR10	file	0.24 KB	2	64 MB	2011-11-16 00:13	rw-r--r--
CDR2	file	0.24 KB	2	64 MB	2011-11-16 00:13	rw-r--r--
CDR3	file	0.24 KB	2	64 MB	2011-11-16 00:13	rw-r--r--
CDR4	file	0.24 KB	2	64 MB	2011-11-16 00:13	rw-r--r--
CDR5	file	0.24 KB	2	64 MB	2011-11-16 00:13	rw-r--r--
CDR6	file	0.24 KB	2	64 MB	2011-11-16 00:13	rw-r--r--
CDR7	file	0.24 KB	2	64 MB	2011-11-16 00:13	rw-r--r--
CDR8	file	0.24 KB	2	64 MB	2011-11-16 00:13	rw-r--r--
CDR9	file	0.24 KB	2	64 MB	2011-11-16 00:13	rw-r--r--

Fig. 2. Hadoop cluster setup

5 Experimental Results

The effectiveness of PIC has been discussed by Lin and Cohen [1]. The scalability of p-PIC has been shown by Weizhong Yan [5]. This paper concentrates on the effectiveness of the algorithm. Experiments where done on various synthetic datasets of varying grid sizes smallest being 2 and largest being 50. The number of iterations and the execution time has been analyzed. The error between the number of iterations is found using Aitken error formula. From the analysis it is found that the number of iterations reduces for faster convergence. The graph for iterations is shown in Fig. 3. Figure 4 shows its execution time with the grid size along the X-axis and execution time along Y-axis. Multinode execution was done with upto 4 slaves and the results were compared. The execution time for multinode setup is show in Fig. 5.

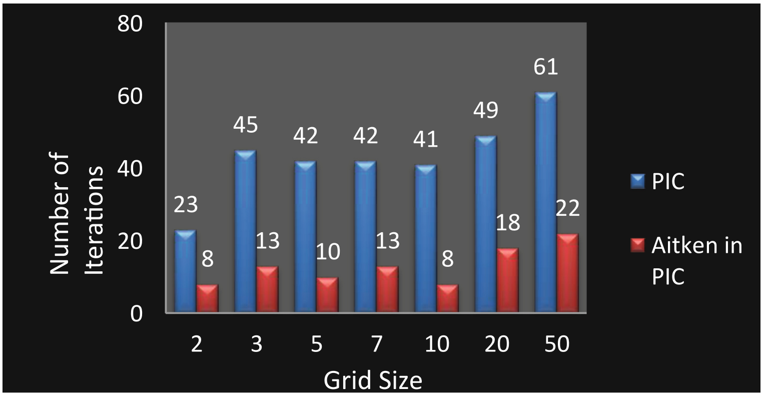


Fig. 3. Number of iterations

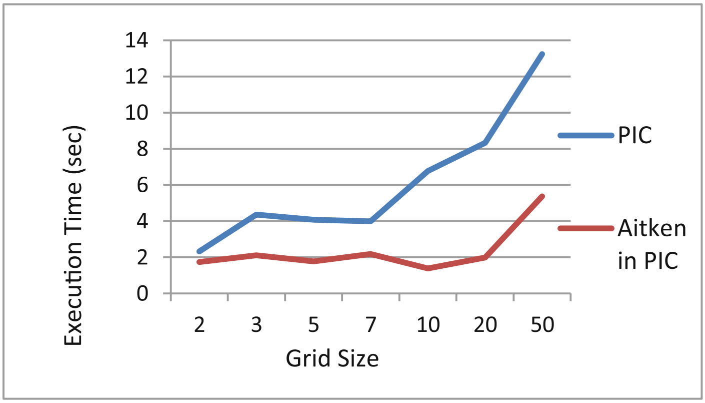


Fig. 4. Execution time for Aitken PIC

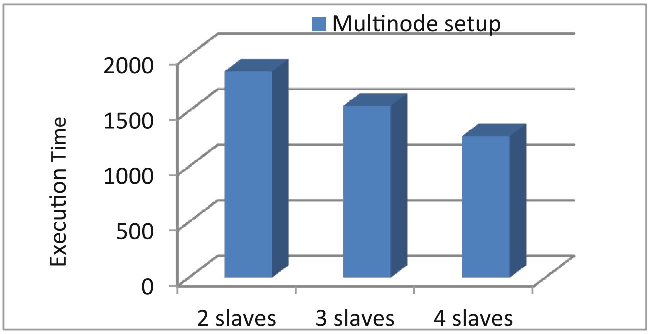


Fig. 5. Execution time in multinode environment

6 Case Study: Clustering Medical Data to Predict the Likelihood of the Diseases

The thriving medical application of clustering in the field of healthcare has helped to group the patients. This increases the quality of hospitals paving a way for better treatment. The analysis has been done efficiently by extracting the medical history of a patient to predict the diseases quickly and accurately. It provides an effectively way to extract information from voluminous heterogeneous data. Algorithm 3 shows the steps involved in it.

6.1 Predictive Analysis System Architecture

There are various phases in this architecture namely Data collection, Feature selection using Questionnaire analysis, Predictive analysis and Report generation. The architectural diagram is shown in Fig. 6.

Data Collection - The datasets are obtained from various sources like health record, personal information, history, laboratories record etc. in various formats and are converted into .csv files. *Feature selection* - Not all of the attributes would help us to predict the disease. Hence feature selection from the dataset collection as done on random basis based on the keyword of importance. *Prediction Analysis* - This analysis is done by the reports generalized by the laboratories and the doctors. The questionnaire session helped more to analyze the cause for the disease. *Report generation* - The graph is generated using Microsoft excel to show the percentage of comparison of the diseases, which helps us to analyze the most affected disease.

```
Map1
    Step 1. m Mappers reads the data and split into<key, value> pairs
    Step2. The corresponding <key, value> list is the input to the map.
    Step 3. Calculate the similarity matrix from the similarity function
      Similarity functio⇦   $s(x_i, x_j)$;  $s(x_i, x_j) = \exp(-\frac{||x_i - x_j||_2^2}{2\sigma^2})$
    Step 4. Find the affinity matrix W=D$^{-1}$A and normalize the similarity matrix W
Reduce 1
    Step 6. Get the intermediate <Key, Value> pairs from the mapper
    Step 7. Calculate the row sum R[i]using norm.
    Step 8. Obtain the initial vector $v^0 = \frac{R}{||R||}$ .
Map2
    Step 9. Obtain the initial vectors from map2.
    Step 10. Choose an integral guess $x_0$,
    Step 11:  Calculate  $p_0$,$p_1$, $p_2$  using any linear  iteration method.
    Step 12:  Compute $\widehat{p_0}$= $p_0$- $\frac{(p_1 - p_0)^2}{p_2 - 2p_1 + p_0}$ and for n=1,2……, and
    Step 13. Compute $p_{n+2}$
    Step 14. Compute $\widehat{p_n}$= $p_n$- $\frac{(p_{n+1} - p_n)^2}{p_{n+2} - 2\, p_{n+1} + p_n}$
    Step 15:  Terminate if | $\widehat{p_n}$- $\widehat{p_{n-1}}$|<  ∈ &  $p \approx \widehat{p_n}$..
Reduce 2
    Step 16. Obtain all vectors from map2.
    Step 17. Emit pair(matrix, vector)
    Step 18. Apply K-Means
Output clusters.
```

Algorithm 3. Aitken power iteration clustering algorithm on MapReduce

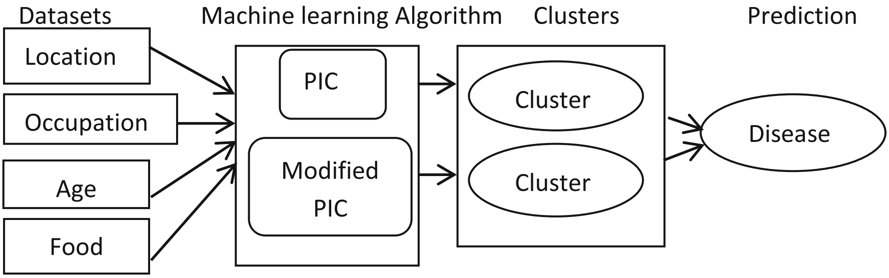


Fig. 6. Prediction system architecture

6.2 Dataset

The data was collected from the primary health center located in our nearby region. It had a history of last 5 years. They include details of nearly 7000 patients. The record contains information of about 8 diagnostic diseases that are being analyzed by siddha treatment. Based on the history of the patient's record the theme of the disease which is the independent variable and age, smoking, location, ancestors, work environment, psychological condition etc. were kept as dependent variables, The patients are clustered based on the diseases and the reason for it is analyzed using regression models.

6.3 Prediction of Eczema Disease

An initial study was done for 600 patients. The study was conducted based on the on the patients answers and part medical history was obtained with the help of the practitioner. These records were used to analyze the disease. Analysis was done to form cluster based on their age, work place, location, food habits. The analysis helped us to predict the disease and treat the patients accordingly. Itchy, red, swollen patches on skin are the symptoms of Eczema disease. Everything in the environment including cigarette smoke, clothes, pollen etc. may be the cause for this disease. Stress also adds to worsen of the disease. Treatment can be divine method or rational. The location, age, job categories and food habits had a major impact.

The results were gathered using the questionnaire section as in Table 1. It was found that a particular age people 30–60 working in an institution dealing with pollens daily are mainly affected by a specific disease called Eczema. Though this disease mostly affects young children than elders it was found that this disease was most common because of the environmental situation and the stress they had on their work. For example the person working as a tutor teaching about the plants was affected by skin problem because of the pollen surrounded near him. Other disease that affected the people where mostly based on medical history.

By conducting this survey the most affected disease a particular location of middle age group was chosen. The reason for their disease was analyzed. The percentage of disease that had effect on the patients is shown Fig. 7. The environmental factors and stress that affected them was noted. Treatment along with counseling reduced the risks

Table 1. Questionnaire categories

Location	Age
✓ Rural – 0	✓ < 30
✓ Urban – 1	✓ 30–60
	✓ > 60
Job categories	Food habits
✓ Agriculture – 1	✓ 3 times – 1
✓ IT – 2	✓ 2times – 0
✓ Banking – 3	
✓ Teaching – 4	
✓ Others – 5	
✓ What is your working hours	
✓ Do you have any problem at work place?	
✓ Does anyone smoke at home?	
✓ Do you own a garden?	
✓ How much time you spend with others	
✓ How much hours do you work daily?	
✓ Habit of eating	

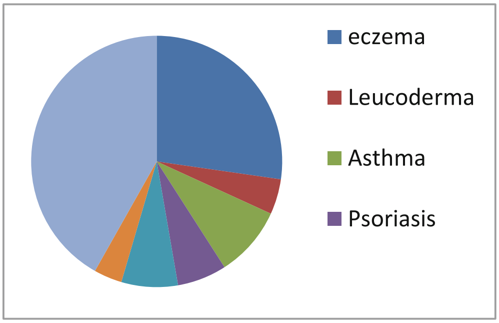


Fig. 7. Analysis of various diseases

of the people affected by 7% over the particular group. It was easy for analysis and prediction. This technique can be applied over Big Data health record and various other disease and their causes can be analyzed and treated accurately.

7 Conclusion

This paper introduces a novel technique to improve the convergence in iterative algorithms. It combines Hadoop and data mining techniques to handle Big Data. The execution results based on evaluation are presented. Big Data analytics using Hadoop paves a good solution for various outcomes that is proved to be best. This work can be extended to predict diseases in general medicine and other health applications. As a future work the algorithm can be tested using GPU for better performance.

References

1. Lin, F., Cohen, W.W.: Power iteration clustering. In: International Conference on Machine Learning, Haifa, Israel (2010)
2. Ng, A., Jordan, M., Weiss, Y.: On spectral clustering: analysis and an algorithm. In: Advances in Neural Information Processing Systems, vol. 14, pp. 849–856. MIT Press, Cambridge (2002)
3. The, A.P., Thang, N.D., Vinh, L.T., Lee, Y.-K., Lee, S.: Deflation-based power iteration clustering. Appl. Intell. **39**(2), 367–385 (2013)
4. Liu, C., Li, Y.: A parallel page rank algorithm with power iteration acceleration. Int. J. Grid Distrib. Comput. **8**(2), 273–284 (2015)
5. Yan, W., et al.: p-PIC: parallel power iteration clustering for big data. J. Parallel Distrib. Comput. **73**(3), 352–359 (2013)
6. Dartmouth: Numerical methods. Chapter 10.3 Power method for approximating eigenvalues, ENGS 91, Dartmouth College, Hanover, New Hampshire
7. Myers, M.E., van de Geijn, P.M., van de Geijn, R.A.: Linear algebra: foundations to frontiers - Notes to LAFF with (2014)
8. Führer, C., Sopasakis, A.: pp. 13–19. http://www.maths.lth.se/na/courses/FMN050/media/material/part3_1.pdf
9. Bumbariu, O.: An acceleration technique for slowly convergent fixed point iterative methods. Miskolc Math. Notes **13**(2), 271–281 (2012)
10. Lambers, J.: Accelerating convergence. Lecture 12 Notes, MAT460/560 (2010)
11. Sidi, A.: Lecture notes on Acceleration of Linear Convergence by the Aitken's Δ^2-Process. Israel Institute of Technology, Israel (2012)
12. Stevens, S.: Numerical analysis. MTHBD423/CMPBD 423, Behrend college, Fall (2003)
13. http://hadoop.apaoche.org/docs/r1.2.1/mapred_tutorial.html
14. http://en.wikipedia.org/wiki/Big_data
15. http://www.cs.uic.edu/~ajayk/c566/HadoopMap-Reduce.pdf
16. Zhengqiao, X., Dewei, Z.: Research on clustering algorithm for massive data based on hadoop platform (2012). 978-0-7695-4719-0/12 $26.00 © 2012 IEEE. doi:10.1109/CSSS.2012.19
17. Dhanapal, J., Perumal, T.: Inflated power iteration clustering algorithm to optimize convergence using lagrangian constraint. In: Silhavy, R., Senkerik, R., Oplatkova, Z., Silhavy, P., Prokopova, Z. (eds.) Proceedings of 5th Computer Science Online Conference. Advances in Intelligent system and Computing, vol. 465, pp. 227–238. Springer, Cham (2016)

Building an Intelligent Call Distributor

Thien Khai Tran(✉), Dung Minh Pham, and Binh Van Huynh

Faculty of Information Technology, Ho Chi Minh City University of Foreign
Languages and Information Technology, Ho Chi Minh City, Vietnam
{thientk, pmdung, binhl601}@huflit.edu.vn

Abstract. This paper presents an intelligent call distributor. This system can route calls to the most appropriate agent according to routing rules constructed by a text classifier. The system includes four main components: a telephone communication network; speech recognition; a text classifier; and a speech synthesizer. To the best of our knowledge, this is one of the first systems in Vietnam to implement an integrated mechanism using both text language processing and spoken language processing. This allows voice applications to be intelligent and to be able to communicate with humans in natural language with high accuracy and reasonable speed. After building and testing, the system is shown to have an accuracy of 92%.

Keywords: Spoken dialog systems · Intelligent call distributor · Voice applications · Voice server

1 Introduction

Voice, as an individual method of expressing and exchanging ideas, is the primary tool of human communication. Based on the idea of voice communication between humans and machines, research on speech recognition as a basis for developing voice applications has attracted much attention in recent decades. Typical voice applications are the voice server solution from IBM, and the system flight information inquiry of CMU. Voice systems can be applied in all areas of social life such as health, education, administration and information provision.

From the 1960s to the 1970s, research was carried out into spoken dialog systems such as ELIZA. However, it was not until the 1990s that voice communication systems allowed a high level of application by integration with interactive systems via telephone such as TRAIN and RAILTEL; currently, IBM's Watson, Iphone's Siri and Microsoft Cortana virtual assistant are the most elite products in voice applications.

In Vietnam, in recent years, the study of speech processing technology has also achieved encouraging results. Many notable publications have been produced by the Institute of Information Technology - Vietnamese Academy of Science and Technology and University of Science, VNU-HCM. It is also worth mentioning the work in [7] as well as [3, 5]. These studies concentrated primarily on improving the efficiency of their voice recognition systems, such as that by Quan Vu et al., which obtained a precision rate of over than 93%. This group successfully built numerous voice applications on this basis. For example, in [5], Quan Vu et al. successfully built the VIS::

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_24

DIR system, in which the caller can speak the names of departments/offices in a university and the system will forward/redirect/route these calls to the appropriate office without requiring a receptionist. However, these applications have not yet been complemented with an efficient text processing mechanism, which is an important mechanism in terms of helping the system to understand commands. It was not until 2014 that the authors of [9] developed an education information inquiry system called EDUvoice. The system is built with four main modules: speech recognition, command processing (text syntactic parsing, text semantic analysis), speech synthesis, and a knowledge base of information on learning. In this system, the handling of the syntax and semantics of the commands in the system are resolved by definite clause grammar [1]. The authors have proposed 17 semantic structures which represent the 48 structures of the text command and can appear in the context of the application. In 2015, in [10], the author also proposed a similar model to query information about hotel reviews. The knowledge base in this system is formed of these reviews, which are classified into two classes of positive and negative using support vector machines [11].

In this paper, we present the Intelligent Call Distributor. This system can identify many types of voice command, convert them into text, and route calls to the most appropriate recipient using routing rules built by the text classifier. The handling of the navigation commands in the system is carried out using the Naïve Bayes machine learning method [12]. Our system also has an automatic speech recognition module and a speech synthesis module. Similarly to our approach in [8–10] we evaluate a Vietnamese speech recognition task using a HTK (Hidden Markov Model Toolkit) [6] and speech synthesis operations using the unit-selection method [2].

We organize the rest of this paper as follows: in Sect. 2, the proposed model is described; in Sect. 3 we report the results of speech processing; in Sect. 4 we present the results of text processing, and finally, we conclude the paper and discuss possibilities for future work.

2 Proposed Model

The system is built using the following functions: identifying calls via telephone; classifying these calls; routing them and then answering the caller. The interaction script between the caller and the system is described below.

Step 0: Listening stage.

Step 1: User speaks to the system.

Step 2: The speech is converted into a Vietnamese text sentence.

Step 3: The system analyzes the text sentence and returns the most relevant choice to the user.

Step 4: The system confirms with the user.

- If the user agrees: Go to Step 5.
- Else: Back to Step 3.

Step 5: The system routes the call to the appropriate agent.

For the purpose of realizing all functions given in the above scenario, the system consists of the following components:

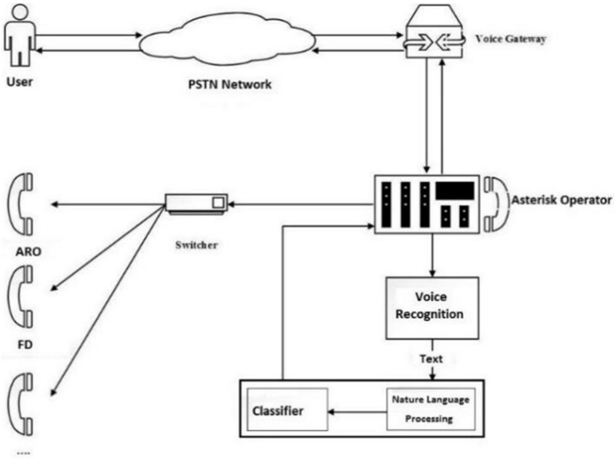


Fig. 1. Architecture of the system

- Asterisk communication: to receive and process signals from the phone, route and answer back to the telephone system.
- Automatic speech recognizer: to identify the words spoken by the user, then convert them into text.
- Text Classifier: to classify the (text) calls for routing to recipients.
- Speech Synthesizer: to convert text to speech.
- Central processor: to connect all modules (Figure 1).

3 Speech Processing

Speech recognition is a process of pattern matching in which the patterns are recognition units, which can be words or phonemes. The fundamental challenge that a speech recognition problem must be treated is the voice varies over time, and there is a large difference between the speech of different individuals. The study of speech recognition is based on three basic principles: (1) the voice signal is represented by spectrum values over a short time frame so that voice features can be extracted and used as data for speech recognition; (2) the content of the speech is expressed as letters which are a series of phonetic symbols. Hence the significance of a pronunciation is preserved when we pronounce the phonetic sequence of acoustic symbols; (3) Speech recognition is a cognitive process. Spoken language has meaning, therefore information about semantics and pragmatics is of value in the speech recognition process, particularly when acoustic information is not clear.

The research field of speech recognition is quite broad and involves many different sectors such as digital signal processing, acoustic, information and computer science theory, linguistics, physiology and applied psychology. Voice recognition systems can be divided into two different categories, discrete and continuous. In continuous speech

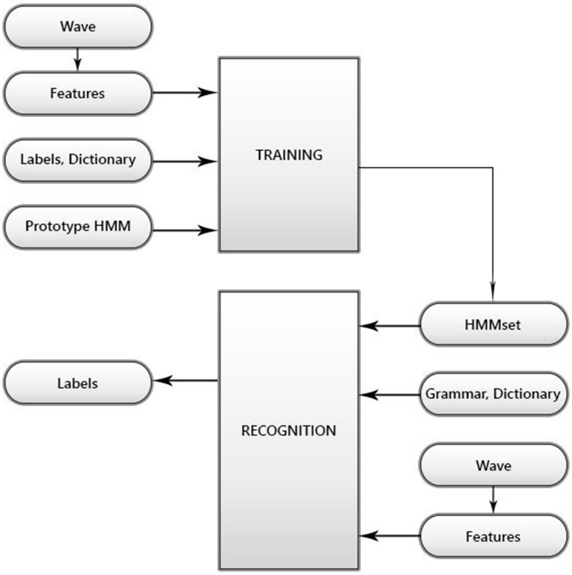


Fig. 2. Steps in building the Automatic Speech Recognizer [8]

recognition, we can again distinguish a small-sized vocabulary recognition system from a medium or large-sized system.

In this system, employing the same approach as in [5, 9–11] we have adopted the HTK Speech Recognition Toolkit [8] to build the speech recognition module. Figure 2 shows the steps necessary to create the Automatic Speech Recognizer with HTK.

The construction of a speech recognizer consists of two stages.
Training stage:

- Prepare the training data;
- Assign labels, build the dictionary;
- Create the HMM prototype for each phone unit.

The output: the set of HMM models that have been trained (hmmset).
Recognition stage:

- Given the HMM models were trained (hmmset);
- Construct grammar;
- Extract voice features.

The output: recognized text.

Training Data

There are 1,450 sentences in the speech corpus, and total audio training took four hours. All speech was sampled at 8,000 Hz, 16 bit in PCM format in a relatively quiet environment, with 40 speakers (25 male and 15 female). The lexicon comprises of 156 words, as shown in Table 1.

Table 1. List of words.

bao	chi	đâu	hai	kỳ	miễn	ngữ	phúc	theo	tốt	xanh
bảng	chỉ	đào	hạn	ký	môn	nguyện	quốc	thẻ	trả	xét
bảo	chính	đại	hết	là	mở	nhận	quy	thi	trễ	yêu
bằng	cho	đăng	hè	làm	mùa	nhiêu	ra	thiệu	trình	tổ
biểu	chuẩn	điều	hoàn	lạc	muốn	như	rèn	thông	trung	chức
biết	chương	điểm	học	lại	mức	những	sách	thời	trường	em
bình	chứng	định	hỏi	lệ	nào	nộp	sau	thu	tuyến	
bộ	chuyên	đối	huỷ	liên	năm	nối	sinh	thực	và	
bống	còn	dục	khảo	lịch	nay	nợ	sĩ	thức	vào	
cầu	công	gặp	khi	lơ	ngành	nữa	tập	tiêu	văn	
cao	có	giấy	khoa	luận	nghệ	ôn	tạo	tin	về	
các	của	gian	khoản	lưu	nghiệp	ở	thành	tích	việc	
cách	cương	giảm	không	luỹ	nghiên	phần	thạc	tín	viên	
chất	cứu	giáo	khoá	luyện	ngoài	phí	thế	tính	vọng	
chế	đầu	giới	kiện	máy	ngoại	phòng	thế	tôi	với	

Grammar

The language model or grammar provides information about the syntactics, semantics, and word order of sentences. This component helps the system select the best identification results from the list of candidates selected by the recognition process. This is a set of constraints which defines the phrases that a speech recognition engine can use to match speech input. HTK also provides a grammar definition format, and an associated HParse tool that can be used to build this word network automatically. We store the grammar definition in a file called gram. a part of its grammar is as follows:

```
$fac = (PHOFNG DDAFO TAJO | KHOA COONG NGHEEJ THOONG TIN | PHOFNG COONG
TASC SINH VIEEN | PHOFNG SAU DDAJI HOJC | PHOFNG TAFI VUJ);
$sen1 = [LAFM OWN | VUI LOFNG] (CHO TOOI | TOOI MUOOSN) (GAWJP | LIEEN LAJC |
NOOSI MASY VOWSI) $fac;
$sen2 = [LAFM OWN | VUI LOFNG] (CHO TOOI | TOOI MUOOSN) [ HORI CASCH | BIEEST
CASCH ] (DDAWNG KYS | HURY | DDOORI) MOON HOJC;
```

Speech Synthesizer

The speech synthesizer is a system that converts free text into speech. This is a process in which a computer reads out the text for a human listener. The speech synthesis can be performed using Formant synthesis or a unit-selection method [2]. With this system, we choose an integrative approach using unit-selection methods, complying process as a summarization in Fig. 3.

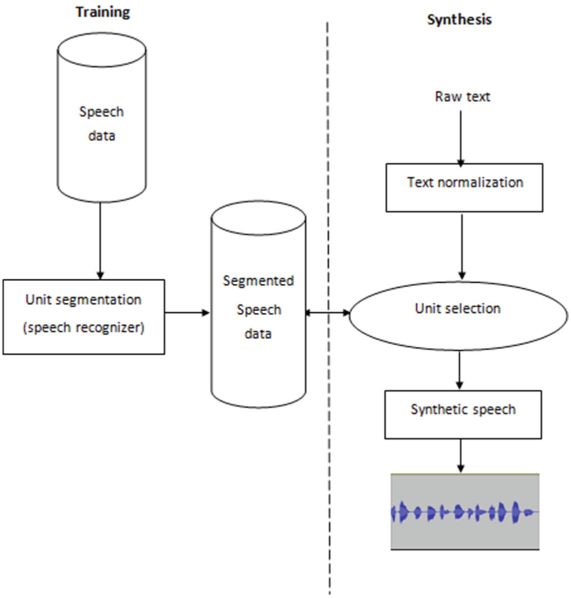


Fig. 3. Speech synthesizer diagram [5]

Evaluation

We make use of the value of Word Accuracy (WA), which is computed as WA = (1 − (S + D + I)/ N) × 100%, to evaluate the performance of the speech recognizer, where:

- N is the total number of words in the testing data,
- S denotes the total number of substitution errors,
- D is the total number of deletion errors,
- and I is the total number of insertion errors.

Table 2. Regression test result by area

Model	Descriptions	Results (accuracy)		
		North	Center	South
VNSE_ A1	Training-only corpus of Northern speakers	95%	75%	92%

Table 3. Regression test result by gender

Model	Descriptions	Result (accuracy)	
		Female	Male
VNSE_ G1	Training-only corpus of male speakers	87%	96%

Performance

The performance test considers four factors: area, gender, age and training corpus. The accuracy of the system is reported in Tables 2, 3, 4, and 5.

Table 4. Regression test result by age

Model	Descriptions	Result (accuracy)	
		18–30	Others
VNSE_G1	Training-only corpus of speakers from 18–30 years old	96%	91%

Table 5. Regression test result by capacity of corpus

Model	Descriptions	Result (accuracy)	
		Trained users	Untrained users
VNSE_C1	Training corpus of 1 speaker	99%	93%
VNSE_C20	Training corpus of 20 speakers	98%	94%
VNSE_C35	Training corpus of 35 speakers	97%	95%

4 Text Processing

Text classification is the process of assigning a natural language document into one or more given categories automatically. This problem was developed in the 1960s and quickly became an issue of research interest for the scientific community and enterprises. A text classifier is used to assist in the process of information retrieval and information extraction, and in decision support systems.

In this report, we try to adopt two machine learning techniques, the Support Vector Machine and Naïve Bayes, to classify the text commands, and then choose a more suitable method. With the classifier, the system can easily route a call to one of the appropriate agents. In this report, we present a case study with five agents as follows: Finance Department (FD); Academic Registrar Office (ARO); Graduate Education Department (GED); Student Service Department (SSD) and Faculty of Information Technology (FIT).

4.1 Support Vector Machine

The support vector machine (SVM) is a statistical model first developed in the mid-1960s by Vapnik. In later years, this model has become one of the most effective machine learning techniques [11]. A SVM performs classification by finding the hyperplane that maximizes the margin between training samples and the hyperplane. In this study, we use LIBSVM3, a library for support vector machines. We collected 2,000 calls corresponding to 40 individuals. The training data was preprocessed and manually labeled as FD, ARO, GED, SSD or FIT. Then, word segmentation and POS tagging were performed thank to [4]. After this, stop-words were removed as well as all

Table 6. The results of the system using the SVM method

Precision	Recall	F-measure
98%	96%	97%

features of which the number of occurrences does not meet a threshold. In this paper, the threshold was set at 2. Finally, the training data set was vectorized and an SVM was used to compute a separating hyperplane.

The contents 200 of calls were randomly collected for testing to evaluate how well the system can identify FD, ARO, GED, SSD or FIT data. The standard Precision, Recall and F-score measures were used. Table 6 shows the results of the system running on test data with threshold = 2.

4.2 Naïve Bayes

Naïve Bayes [12] is a simple classifier based on the Bayes theorem. The main idea of the Naïve Bayes approach in classifying documents is the use of conditional probabilities between words and topics to predict the topic of a text that needs classifying. The most important point of this method is the assumption that the all the words in a document appear independently of each other.

Using Naïve Bayes, the following steps were taken to predict a document as being FD, ARO, GED, SSD or FIT:

- A search of the corpus for words that appear in the document;
- A calculation of the probability of the document being FD, ARO, GED, SSD or FIT;
- A comparison of the five probabilities to choose the highest value.

Table 7 shows the results of the system running on test data, as described in Sect. 4.1.

Table 7. The results of the system using the Naive Bayes method

Precision	Recall	F-measure
96%	94%	95%

The Naïve Bayes machine learning model was used for the text classifier of our system for the following reasons:

- Although Naive Bayes does not outperform SVM, its processing time is shorter. This is necessary for the system to interact directly with the user.
- The classification result of the Naïve Bayes method is based on the highest probability. This property gives many options for classification results up to the probability priority.

5 Experiments

In this section, we report the results of experiments on the system, as well as performing an assessment of users of the system, including the speech synthesis component.

5.1 System Experiments

The environments for these experiments are described in Table 8.

The system correctly executes 92 of 100 calls, and thus demonstrates an accuracy of 92%. The average feedback time of the system is about 2.4 s for each command.

Table 8. Experimental environments

Number of calls	Environment	Sampling rate	Quantization	Format
100	in-door	8 kHz	16 bits	PCM

5.2 Investigation

A survey was also conducted of 100 callers who used the system, using the question "Is this system easy to use or not?" There were four levels of evaluation, and the results are shown in Fig. 4.

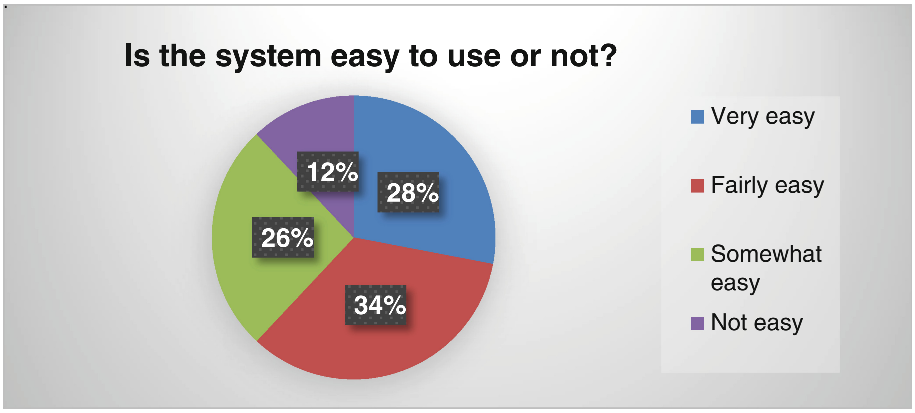


Fig. 4. System investigation

6 Conclusion

This paper is one of the first examples of a Vietnamese voice application that integrates a natural language processing mechanism. We believe that this research, which is a combination of spoken language and written language processing, will enable the development of similar applications. Future work will expand our research with more vocabulary. We will also consider trying to adopt some other machine learning methods to help the system become more robust.

Acknowledgment. This work is supported by Ho Chi Minh City University of Foreign Languages and Information Technology (HUFLIT).

References

1. Pereira, F.C.N., Shieber, S.M.: Prolog and Natural-Language Analysis, pp. 1–284. Microtome Publishing, Brookline (2005)
2. Hunt, A.J., Black, A.W.: Unit selection in a concatenative speech synthesis system using a large speech database. In: Proceedings of ICASSP 1996, vol. 1, pp. 373–376 (1996)
3. Pham, N., Vu, Q.: A spoken dialog system for stock information inquiry. In: Proceedings of IT@EDU, Ho Chi Minh City, Vietnam (2012)
4. Phuong, L.-H., Thi Minh Huyên, N., Roussanaly, A., Vinh, H.T.: A hybrid approach to word segmentation of vietnamese texts. In: Martín-Vide, C., Otto, F., Fernau, H. (eds.) LATA 2008. LNCS, vol. 5196, pp. 240–249. Springer, Heidelberg (2008). doi:10.1007/978-3-540-88282-4_23
5. Vu, Q., Le, C.: Voice server and its applications. Technical report, Research project, HCM City Department of Science and Technology, Vietnam (2012)
6. Young, S., et al.: The HTK Book (version 3.4) (2006). www.htk.eng.cam.ac.uk/docs/docs.shtml
7. Vu, T., Luong, M.: The development of Vietnamese corpora toward speech translation system. RIVF-VLSP 2012, Ho Chi Minh City, Vietnam (2012)
8. Tran, T.K., Nguyen, D.T.: Semantic processing mechanism for listening and comprehension in VNSCalendar system. Int. J. Nat. Lang. Comput. (IJNLC) **2**(2), 1–15 (2013)
9. Tran, T.K., Tran, T.C.K., Mai, T.A., Nguyen, N.M.H., Vu, H.T.: EDUVoice - a system for querying academic information via PSTN. In: The Third Asian Conference on Information Systems (ACIS 2014), Nha Trang (2014)
10. Tran, T.K.: SentiVoice - a system for querying hotel service reviews via PSTN. In: IEEE-RIVF 2015, Can Tho, Vietnam (2015)
11. Vapnik, V., Cortes, C.: Support-vector networks. Mach. Learn. **20**, 273–297 (1995)
12. Friedman, N., Geiger, D., Goldszmidt, M.: Bayesian network classifiers. Mach. Learn. **29**(2–3), 131–163 (1997). Special issue on learning with probabilistic representations

A Fuzzy Logic Based Recommendation System for Classified Advertisement Websites

Umar Sharif, Md. Rafik Kamal, and Rashedur M. Rahman(✉)

Department of Electrical and Computer Engineering, North South University,
Plot-15, Block-B, Bashundhara, Dhaka 1229, Bangladesh
{umar.sharif, rashedur.rahman}@northsouth.edu,
rafik.rkn@gmail.com

Abstract. Classified e-commerce sites have seen a rapid growth in the last few years with the availability of internet to mass people. But as most sites do not offer an intelligent recommender system, ordinary customers looking for a specific product face the daunting task of finding the perfect product in accordance with his requirements and budget. Besides, most of the time, sellers do not have the idea about the exact market value of their item. It results in either under valuation or over valuation which hampers to get a good price that could benefit both the seller and the buyer. We propose a fuzzy logic based intelligent recommender system which will intelligently recommend products most suitable with buyer's requirements. It does not require extensive user information. Though we have used it only for mobile devices in the experiment, results indicate that the system is effective and efficient, and can be implemented for any product based on their features.

Keywords: Fuzzy logic · Recommendation system · Near fuzzy compactness · e-commerce websites

1 Introduction

E-commerce sites have brought the market to the door step of everyone. Anyone can now visit the websites and has access to various categories of products in an instant. The market is now competitive. Besides, the number of offered goods are so huge that it is difficult to keep up the with the satisfaction level of service to the large number of growing consumers. The customers cannot go through the list of all products through the computer devices. On the other hand, the sellers cannot attract the attention of their customers in spite of having the goods. Besides persons or companies who can reach to the customers might not have the product that would attract those specific customers. So, they are losing a huge market. To avoid this, the e-commerce websites have adopted some business strategies to reach every single customer who visits their websites. They are now keeping an eye on the customer preference and purchase history so that they can provide them with the products of their need. This mammoth task of satisfying every customer's need has given birth to the concept of recommendation system [1].

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_25

Now e-commerce sites are concentrating to provide their customers with better service to meet their customers' need. The websites are keeping track of the customer's interaction with products in the websites. They keep the information in their back-end data storage. When the same individual visits the website, the system shows the user the products of his/her liking or previously purchased. This has given a huge boost in selling of products and also reached a satisfaction level of customers to such an extent that now the sellers are getting a group of loyal customers.

A good personalization system provides an easy access to the items of an individual customer likes. Such a marketing strategy is developed to keep up with the customer's satisfaction. The system keeps track of every customer's purchase and browsing history. With that detailed information, the system develops a technic that finds the products that would be similar to the liking of the customer's taste. Thus, the interaction frequency with customer rises. It gives more opportunity of transaction and the benefits goes to the internet enterprises [2, 3].

In this research, we use the fuzzy logic for choosing electronic products like mobile phones which have now become a part and parcel of everyday life. There is a huge market for both new and used mobiles in Bangladesh. In this research, we not only want to make an interactive system which is optimal but also is user friendly. Here a prospective customer can search his/her desired device according to his required preferences. This method of reaching products to the customers not only promotes the rise in business sale but also improves the customer and seller interaction which is vital for modern business market.

2 Related Works

Weng and Liu [4] argued that interacting with the customers was the best way for personal recommendation. Understanding the customer's requirement helps to intelligently recommend products. Li and Liu [5] showed that maintaining iterative customer profile would help to provide the most appropriate information. They also showed the possibly to select optimal products based on customer's preference. Schifer et al. [1] observed the use of recommender systems in e-commerce sites and noticed their contribution in increasing the sales. Based on that, they created a list of requirements such as inputs from the consumers, presentation of recommendation ways to the customer, use of technology and the amount of personalization used in recommendation. Huang and Huang [6] addressed the possibility to use multiple methods to intelligently use the web. As it was noticed that content-based, collaborative filtering, web mining and other current recommendation systems were not adaptive, they introduced a personalized recommendation system and explained its components. This system incorporated all the existing methods to find the most optimal products for the user. When there is lack of user profile data, user behavior on web can be used to predict user's intentions. Cho, Jeong and Lee [7] gave different weight to different components of the product attributes and used algorithms to improve the efficiency. In another paper [8], Devnath and Ganguly proposed the hybridization of collaborative and content based filtering. Attributes were weighted according to their importance which was derived from regression equation originating from the social network behavior of the user. Rather than content based recommendation, all the articles were given with some keywords. Matching those keywords among various articles a relationship was established to recommend the user the desired article [9]. In this paper [10], the idea of fuzzy near compactness was utilized to establish a connection between the customer requirement and the desired product condition. Products with least amount of compactness with regard to the customer's preferences were selected as optimal choice. There was also a similar method used in the paper [11] to recommend multi featured products to customers based on their needs.

Most of the recommendation systems used by current e-commerce sites are based on collaborative filtering which requires a large amount of existing user data to compare with other users. Without the data, they are mostly inefficient because of cold start. But this system requires no user data except the requirements provided by the customers which makes it much more accurate. Even in content based filtering which does not rely on user data, ratings are needed to compare with products. But the subset of rated products is very small compared to overall users and products. This system is rating independent which makes it efficient than both filtering methods. Besides, we choose fuzzy logic because of its efficiency. As there are billions of products and users, assessing all of them would require tremendous amount of computing power. But the fuzzy based recommender system is simple yet very effective. It requires less computation time compared to other methods.

3 Description

3.1 Recommendation System

Customer satisfaction creates a marketing environment where the websites get loyal purchasers and a remarkable escalation in selling. Recommendation system is the solution to meet the demand of the modern e-commerce trading culture [12]. Personalization is used to promote products to customers based on the basis of customer interest and previous interactions in the recommendation system [13]. Separate methods can be utilized to impart personalized information services in a personalization process.

In the first method, the personal preference of the customers is accumulated by the system. Then the personalization system stores the data using sophisticated method and processes the proclivity of the customers by analyzing the gathered customers' preferred information. After the preference data of an individual is stored in the system, a computational model is generated for that individual. This model provides a projection of preference for that certain individual in an explicit domain of products. This method works very well for high-frequently-purchased products in fixed domains.

The second method is based on particular requirement of customers on certain domains. This is basically useful for less-frequently-purchased products. In this case the customer should have sufficient knowledge of the domain of the product that he/she wants to purchase. Using that knowledge, the consumer assesses the quality of the desired product. For this method, there is no need to store end-user preference information because the system takes preference of the product directly. Then the system manipulates the product preference information supplied by the customer and suggestions are generated in an optimal manner. So basically, the system provides the customer result on some specific product. This helps the customer to have an idea what he/she actually wants, verify different choices and come to make a final decision. Here all the products are stored beforehand in the database with some certain criteria. Then the preference of the customer and the criteria of the product are matched using sophisticated modeling procedure and suggest the target products. The customer can change his/her preference and the system suggests new products according to that. Gradually the customers can find their desired product.

3.2 Personalization

Personalization is a technique by which a service or a specific product is fit with the desired specific requirement of a specific and unique individual. With this, the idea of bringing the goods to the customers' door using the internet reaches a new height. Through this method e-commerce sites can monitor customer choices, searching and purchasing behavior. Accordingly, they develop appropriate business strategies to represent their products to their customers of some certain type and try to inform them with sufficient information which will suit their likings and delivering the products. As a result, customers are satisfied with the service. Besides, this strategy boosts the satisfaction level and increases the searching frequency of the customers to a great extent. The e-commerce sites get a loyal group of customers and an expansion of business with eloquent benefits [6, 7].

In this paper, we use the concept of personalization. We store the necessary data of the products and process the data from storage bank when required. Also the unique desired conditions and constraints taken from the end users are used to initiate the interacting procedure to display the outcome. Here the data from the end user and the product information is manipulated to generate a personalization method based on product domain. Customers here can provide information through electrical devices such as computer, laptop, android phones etc. to establish an interaction between them and the system.

3.3 Fuzzy Logic System

A fuzzy logic system utilizes the concept of fuzzy logic using nonlinear mapping of crisp data set and generates scalar data set as output [11]. A fuzzy logic system is structured with four primary components: a fuzzifier, a rule base, an inference engine and a defuzzifier. A crisp set of data as input are transformed into a fuzzy set utilizing fuzzy linguistic variables, fuzzy linguistic terms, and membership functions. This conversion of crisp set to fuzzy set is known as Fuzzification. Then an interface is generated using some given set of formulas and gives a set of fuzzy numbers. Finally, the resulted fuzzy output set is converted to a set of crisp output using membership functions in the defuzzifier also known as Defuzzification [8].

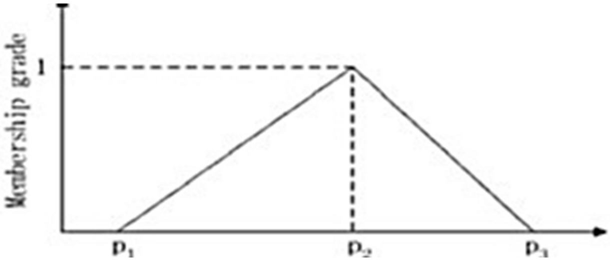


Fig. 1. Membership function for a fuzzy number

We use the notion of Triangular Fuzzy Number (TFG) for expressing customer's requirements and various product attributes. Triangular Fuzzy Number is a fuzzy set that has a membership function in the shape of a triangle to view the likely distribution of fuzzy number. Assuming that T is a Triangular Fuzzy Number, then T = T1, T2, T3 where T1, T2, and T3 are crisp numbers with condition T1 $\leq$ T2 $\leq$ T3. Figure 1 provides an example of TFG. For example, if we have a Triangular Fuzzy Number, C = (1,2,3) then C1 = 1, represents the left most point, C2 = 2 represents the pick and the central point and lastly C3 = 3 represents right most point. TFG can be represented as linguistic variables as such for temperature like 'Low' as T(1,2,3), 'Medium' as (2,3,4) and 'High' as (3,4,5). We proposed TFG for 'condition' in Table 1 and 'use duration' in Table 2.

Table 1. Linguistic variable of condition

Linguistic term	Triangular numbers
Defective	(1,2,3)
Slightly defective	(2,3,4)
Working	(3,4,5)
Good	(4,5,6)
Excellent	(5,6,7)

Table 2. Linguistic definition of use duration

Linguistic term	Triangular numbers
Unused	(1,2,3)
Less than 3 months	(2,3,4)
3–6 months	(3,4,5)
6–12 months	(4,5,6)
1 to less than 2 years	(5,6,7)
More than 2 but less than 3 years	(6,7,8)
More than 3 years	(7,8,9)

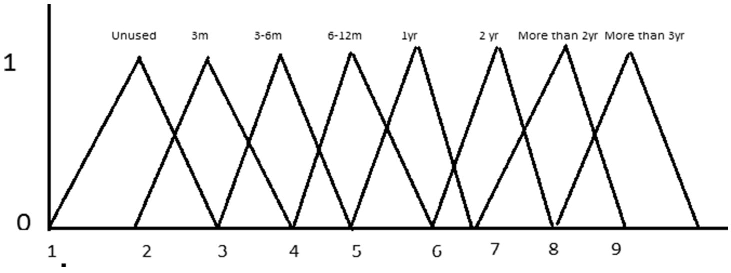


Fig. 2. Membership function for 'use duration'

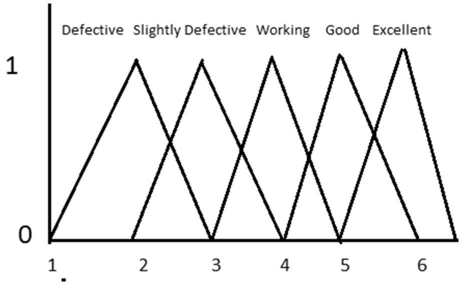


Fig. 3. Membership function for 'condition'

We have 5 linguistic terms for 'condition' (Defective, Slightly Defective, Working, Good, and Excellent) as stated in Table 1 and showed in Fig. 3. Each has a TFG value for example, Good has a TFG value (4,5,6). The range of TFG value is 1–7. For 'Use Duration', there are 7 variables (Unused, Less than 3 months, 3–6 months, 6–12 months, 1 to less than 2 years, More than 2 years, More than 3 years). The range of TFG here is 1–9. Those are shown in Fig. 2.

4 Methodology

Methodology is described in the following sections.

4.1 System Architecture

System architecture uses a combination of data mining technology and fuzzy logic system. An interactive simple interface has been built to assist the seller and buyer. When a seller wants to sell a product, he defines the type of product to categorize. As stated before, every product is unique in its features and conditions. A product may be in a good shape or maybe a bit up used. How longer a product has been used is also another defining factor for the price. For these reasons, the seller mentions the specific

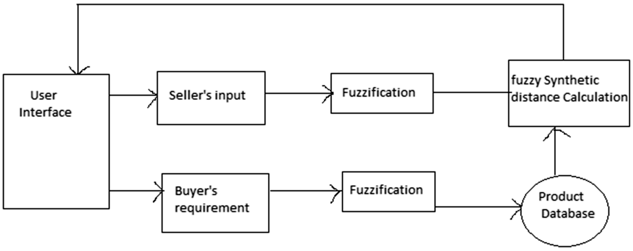


Fig. 4. System architecture

state of his item and selects the conditions. The system transforms and fuzzifiers the condition according to the database. When a buyer needs to find specific product, he requires an intelligent recommender which will help to specify his requirements. Not all people have similar need even for a similar product. For example, a person buying an iPhone 6 may not need a completely new one. If his budget is limited, his satisfaction level depends on the model rather than the condition or use duration. So, the customer defines his budget and minimum requirements as per expectations. The system fuzzifies the conditions. His requirements are then adjusted with the given product descriptions using the distance calculation and co-efficient calculation methods. The system consists of mainly 4 modules: (i) seller's product conditions (ii) buyer's requirement features (iii) weight factors (iv) product recommendation windows. System architecture is presented in Fig. 4.

The implementation procedure is used in the following order:

1. Implement the algorithm for transforming the product's condition into fuzzy membership functions.
2. Implement the algorithm for transforming the customer's requirements into fuzzy membership functions.
3. Implement the algorithm to evaluate the seller's requirements with each product's features using fuzzy geometrical distance and fuzzy synthetic distance methods.
4. Use the weight factors to search for optimal product matches.
5. Recommend products with best similarity to customer's need.

4.2 Identifying and Weighting Product

The process starts with the potential seller's input. Seller registers his intended product in the database. Now similar products might be sold by hundreds of sellers but all products are not of same value. Some products are less used and some are used for a longer period of time. Naturally the less it is used, the more value it has. Similarly, product also plays an important role. The better the condition of the product, more price people are willing to pay for it. But not everyone requires brand new product.

Some requires product albeit at a premium price but someone looks for ones in cheap price. But generally, while searching for a product, all kinds and priced items are suggested at once. To optimally suggest products, buyer's requirement needs to be matched with the products in database. That is why the seller first registers his products in the database.

The database containing *'n'* number of products which is represented by the vector,

$$\mathrm{Product_i} = (\mathrm{procon_i},\ \mathrm{produr}_i) \text{ where } i = 1\ldots n. \tag{1}$$

In vector (1), procon is abbreviation for product condition, produr for product duration. Different options for conditions are given according to Table 1. The user's input is transformed into a Triangular Fuzzy Number.

$$\mathrm{Procon_i} = \left(\mathrm{C_i^1},\ \mathrm{C_i^2}, \mathrm{C_i^3}\right) \text{ where product number, } i = 1\ldots n \tag{2}$$

Different options for use durations are given according to Table 2. The user's input is transformed into a Triangular Fuzzy Number.

$$\mathrm{Produr_i} = \left(\mathrm{C_i^{1\sim}},\ \mathrm{C_i^{2\sim}}, \mathrm{C_i^{3\sim}}\right) \text{ where product number, } i = 1\ldots n \tag{3}$$

This way all the products of similar category are put into the vector.

The value of the TFG is transformed according to fuzzy rule described in Tables 1 and 2.

For example, if a mobile phone is for sale, suppose the condition is set to 'working' and the use duration is set to '6–12 months'.

So, from Table 1, we can set the value to

$$\mathrm{proconmobile} = (3{,}4{,}5) \tag{4}$$

In vector (4), proconmobile is abbreviation for product condition of mobile.

From Table 2, we can write,

$$\mathrm{Produrmobile} = (4{,}5{,}6) \tag{5}$$

where produrmpbile is abbreviation for product duration of mobile.

So the vector, Product$_1$ = (proconmobile, produrmobile).

Similarly, when a buyer is interested buy a product, he selects his choice about product condition and use duration which saved in the vector.

$$\mathrm{Custom} = (\mathrm{cuscon},\ \mathrm{cusdur}) \tag{6}$$

In vector Custom (6), cuscon is customer's given condition, cusdur is customer's given duaration.

Here, cusdur $= (\mathrm{c1},\ \mathrm{c2}, \mathrm{c3})$ is a TFG and cuscon $= \left(\mathrm{C}_1^{\sim}, \mathrm{C}_2^{\sim}, \mathrm{C}_3^{\sim}\right)$ is also a TFG.

Since we will not use collaborative filtering, a database for buyer's choice is not necessary.

4.3 Mapping Buyer's Requirements with Products

4.3.1 Euclidian Near Fuzzy Compactness

After getting $\mathrm{produr_i}$ for all the products and customer requirement vector Custom, we proceed to compute the Euclidian Fuzzy Near Compactness between $\mathrm{produr_i}$ and cusdur as defined in the following equation,

$$\mathrm{N_{duration}}(\mathrm{produr_i}, \mathrm{cusdur}) = \sqrt{(}\sum\nolimits_{j=1}^{3} (\mathrm{C_i^j} - \mathrm{Cj})^2) \text{ Where } i = 1\ldots n. \tag{7}$$

Similarly, we proceed to compute the Euclidian Fuzzy near Compactness between $\mathrm{procus_i}$ and cusdur as defined in the following equation,

$$\mathrm{N_{condition}}(\mathrm{procon_i}, \mathrm{cuscon}) = \sqrt{(}\sum\nolimits_{j=1}^{3} (\mathrm{C_i^j} - \mathrm{Cj}^{\sim})^2) \text{ Where } i = 1\ldots n. \tag{8}$$

4.3.2 Manhattan Distance Measurement

The Manhattan distance measures the distance between two points in a grid-like path. The Manhattan distance between two items is the sum of the differences of their corresponding components.

Here, the distance between product's condition and potential buyer's required condition is measured by Manhattan distance.

$$\mathrm{D_{condition}}(\mathrm{procon_i}, \mathrm{cuscon}) = \sum\nolimits_{j=1}^{3} |\mathrm{C_i^j} - \mathrm{Cj}^{\sim}| \text{ where } i = 1, 2, \ldots n. \tag{9}$$

The distance between product's 'use duration' and potential buyer's required 'use duration' is measured by Manhattan distance as followed,

$$\mathrm{D_{duration}}(\mathrm{produr_i}, \mathrm{cusdur}) = \sum\nolimits_{j=1}^{3} |\mathrm{C_i^j} - \mathrm{Cj}| \text{ where } i = 1, 2, \ldots n. \tag{10}$$

4.3.3 Weighting and Finalizing the NFC and Manhattan Distance

Since the product condition and duration has different impact on buyer's and seller's choices, we provide a weight to the fuzzy near compactness values.

So, the final NFC value,

$$\mathrm{TFGNFC} = \mathrm{v_1} * \mathrm{N_{duration}} + \mathrm{v_2} * \mathrm{N_{conditin}} \tag{11}$$

We consider $\mathrm{V_1} = 0.4$ and $\mathrm{V_2} = 0.6$.

$$\begin{aligned}&\text{NFCE (Near Fuzzy Compactness by Euclidian Distance)}\\ &\quad = 0.4 * \mathrm{N_{duration}} + 0.6 * \mathrm{N_{condition}}\end{aligned} \tag{12}$$

Similarly,

$$\text{NFCM (Near Fuzzy Compactness by Manhattan Distance)} = 0.4 * \mathrm{D}_{\text{duration}} + 0.6 * \mathrm{D}_{\text{condition}} \quad (13)$$

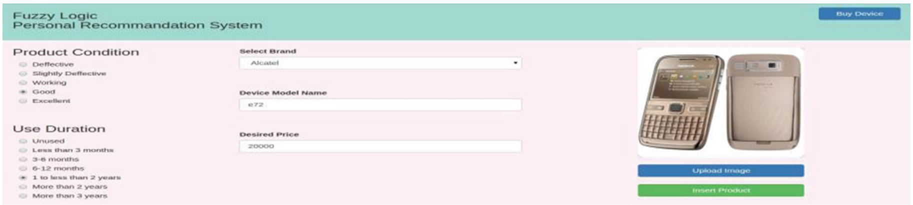
5 Result and Implementation

The proposed system can be applied to recommend optimal products which will be useful for potential buyers lessening their time and effort in buying products. Therefore, the experiment emphasized on the implementation of the system with an objective to recommend desirable products precisely.

5.1 Dataset

Data about 150 cellphones were collected from Bikroy.com. Since most of the data is in Bengali, they were manually inputted in the system database.

In Fig. 5, the interface to input the product's features is shown. Features such as brand, model, capacity etc. are used to categorize each phone for precise recommendation.



Fuzzy Logic
Personal Recommandation System

Product Condition

○ Deffective
○ Slightly Deffective
○ Working
● Good
○ Excellent

Use Duration

○ Unused
○ Less than 3 months
○ 3-6 months
○ 6-12 months
● 1 to less than 2 years
○ More than 2 years
○ More than 3 years

Select Brand: Alcatel

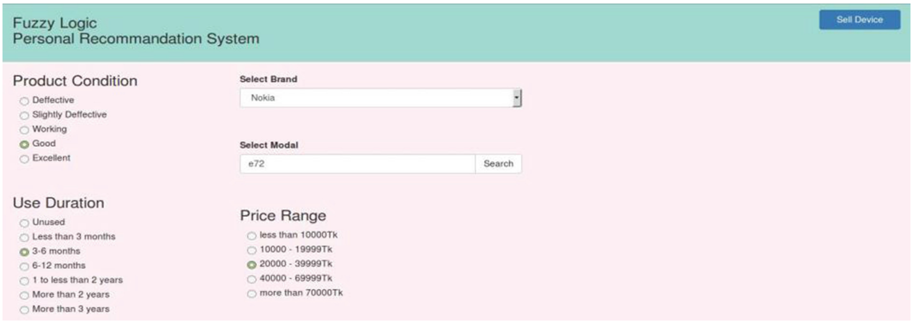
Device Model Name: e72

Desired Price: 20000

Fig. 5. Product seller's window

Product information is stored in the following order:

(i) The seller selects his phone brand from the list of different phone manufacturers.
(ii) After selecting the brand, the seller is prompted to enter the model number on text box and it's stored in the database.
(iii) The desired price of the phone is mentioned. The seller can also upload the picture of the device.
(iv) Then the seller selects the specific product condition among the 5 choices for 'condition' (Defective, Slightly Defective, Working, Good, Excellent).
(v) After that, the duration of use is selected from among the given different time durations (Unused, Less than 3 months, 3–6 months, 6–12 months, 1 to less than 2 years, More than 2 years, More than 3 years). Gradually, the entire information about cell phones is stored in a database.



Fuzzy Logic
Personal Recommandation System

Product Condition
- ○ Deffective
- ○ Slightly Deffective
- ○ Working
- ● Good
- ○ Excellent

Use Duration
- ○ Unused
- ○ Less than 3 months
- ● 3-6 months
- ○ 6-12 months
- ○ 1 to less than 2 years
- ○ More than 2 years
- ○ More than 3 years

Select Brand: Nokia

Select Modal: e72

Price Range
- ○ less than 10000Tk
- ○ 10000 - 19999Tk
- ● 20000 - 39999Tk
- ○ 40000 - 69999Tk
- ○ more than 70000Tk

Fig. 6. Buyer's window

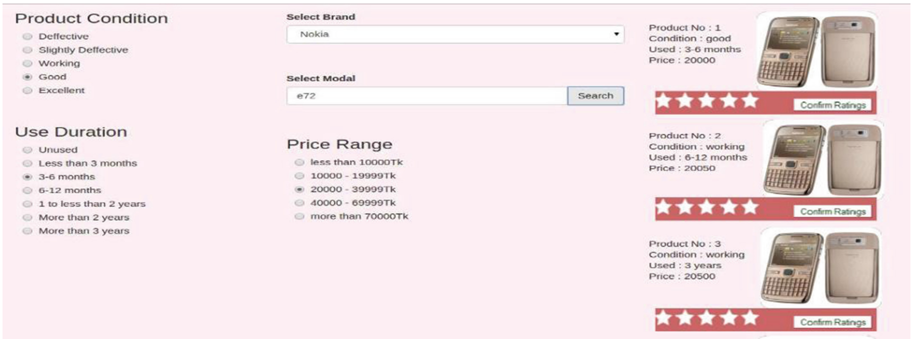
In Fig. 6, the potential buyer's requirement input window is shown. The buyer's course of action is as followed,

(i) Buyer first selects his desired brand and mentions the model number.
(ii) Then he selects the desired 'Product Condition' from (Defective, Slightly Defective, Working, Good, and Excellent).
(iii) Preferred 'Use Duration' is chosen from (Unused, Less than 3 months, 3–6 months, 6–12 months, 1 to less than 2 years, More than 2 years, More than 3 years).

Finally, the price range is selected. Then the buyer presses the 'search button' for recommended products.

5.2 Result

The Fig. 7 shows the recommended products for the consumer's need. 10 phones are recommended by measuring between the synthetic compactness between the buyer's requirements and the product features specified by the seller. Only the products of same brand and model numbers are recommended.



Product Condition
- ○ Deffective
- ○ Slightly Deffective
- ○ Working
- ● Good
- ○ Excellent

Use Duration
- ○ Unused
- ○ Less than 3 months
- ● 3-6 months
- ○ 6-12 months
- ○ 1 to less than 2 years
- ○ More than 2 years
- ○ More than 3 years

Select Brand: Nokia

Select Modal: e72

Price Range
- ○ less than 10000Tk
- ○ 10000 - 19999Tk
- ● 20000 - 39999Tk
- ○ 40000 - 69999Tk
- ○ more than 70000Tk

Product No : 1
Condition : good
Used : 3-6 months
Price : 20000

Product No : 2
Condition : working
Used : 6-12 months
Price : 20050

Product No : 3
Condition : working
Used : 3 years
Price : 20500

Fig. 7. Recommended product list

5.3 Evaluation

To evaluate the performance of the system, a customer satisfaction rating system is introduced. Each recommended product has 1 to 5 rating option where 1 means the buyer is least satisfied and 5 means the buyer is fully satisfied with the recommendation. The customer can give stars for rating, for example, if he is fully satisfied he can give 5 star rating. 15 persons from different backgrounds (Engineering, Business, Art etc.) evaluated the system and rated their satisfaction level for each recommended product. The average rating of 10 users on different products is given in Table 3.

Table 3. Average rating by users

User	Rating (R_i)
User 1	4.7
User 2	4.2
User 3	4.6
User 4	3.9
User 5	4.8
User 6	4.8
User 7	4.2
User 8	4.5
User 9	4.6
User 10	4.8

Now, we measure the efficiency of the system.

$$\text{Efficiency} = (1 - \frac{\sum_{i=1}^{i=10}(5 - Ri)}{50}) * 100\% \tag{14}$$

The system has an efficiency of 90.2% which is satisfactory and can be improved further.

6 Future Work

Right now, the system is built only for recommending cellphones. But it can be used for products of almost everything from cars, home appliances, and electronic products to even houses. But it needs specific feature set for each domain of products. In future, we will expand the system for all these products as well. Besides, the range and scope of the membership function will be adjusted for better accuracy. Since different products have different features and unique conditions, new membership functions will be added. The system will have sophisticated interaction based interface to simplify the user's experience. The system can be further developed into mobile application for successful transformation into a commercial venture.

References

1. Schafer, J.B., Konstan, J., Riedi, J.: Electronic commerce recommender applications. J. Data Min. Knowl. Disc. **5**(1–2), 115–152 (2001)
2. Resnick, P., Varian, H.: Recommender systems. Commun. ACM **40**(3), 56–58 (1997)
3. Chen, D.N., Paul, J.H.H., Kuo, Y.R., Liang, T.P.: A web based personalized recommendation system for mobile phone selection: design, implementation, and evaluation. J. Expert Syst. Appl. **37**(12), 8201–8210 (2010)
4. Weng, S.S., Liu, M.-J.: Personalized product recommendation in e-commerce. In: Proceedings of the 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service, pp. 413–420 (2004)
5. Lee, W.P., Liu, C.H.: Intelligent agent-based systems for personalized recommendations in internet commerce. Expert Syst. Appl. **22**(4), 275–284 (2002)
6. Huang, L., Dai, L., Wei, Y., Huang, M.: A personalized recommendation system based on multi-agent. In: Second International Conference on Genetic and Evolutionary Computing, WGEC 2008 (2008)
7. Cho, H., Jeong, O.-R., Lee, E.: A personalized recommendation system based on product-specific weights and improved user behavior analysis. In: Proceedings of the 4th International Conference on Uniquitous Information Management and Communication, ICUIMC, Article No. 57 (2010)
8. Debnath, S., Ganguly, N., Mitra, P.: Feature weighting in content based recommendation system using social network analysis. In: Proceedings of the 17th International Conference on World Wide Web, pp. 1041–1042 (2008)
9. Zahiduzzaman, A.K.M., Quasem, M.N., Ahmed, F., Rahman, R.M.: Indexing Bangla newspaper articles using fuzzy and crisp clustering algorithms. In: Proceedings of 13th International Conference on Enterprise Information Systems (ICEIS 2011), vol. 1, pp. 361–364 (2011)
10. Mendel, J.: Fuzzy logic systems for engineering. A tutorial. Proc. IEEE **83**(3), 345–377 (1995)
11. Tsai, H.C., Hsiao, S.-W.: Evaluation of alternatives for product customization using fuzzy logic. J. Inf. Sci. **158**, 233–262 (2004)
12. Schafer, J.B., Konstan, J., Riedi, J.: Recommender systems in e-commerce. In: Proceedings of the First ACM Conference on Electronic Commerce (1999)
13. Mobashe, B., Dai, H., Luo, T.: Discovery and evaluation of aggregate usage profiles for web personalization. Data Min. Knowl. Disc. **6**(1), 61–82 (2002)

Potential Risk Factor Analysis and Risk Prediction System for Stroke Using Fuzzy Logic

Farzana Islam, Sarah Binta Alam Shoilee, Mithi Shams,
and Rashedur M. Rahman(✉)

Department of Electrical and Computer Engineering, North South University,
Plot-15, Block-B, Bashundhara, Dhaka 1229, Bangladesh
farzana77sweety@gmail.com, sarahshoilee121@gmail.com,
shamsmithi@yahoo.com, rashedur.rahman@northsouth.edu

Abstract. Stroke is a life-threatening, deadly cause, which occurs due to the interruption of blood flow to any part of brain. As stroke is a globally alarming deadly cause, using computational expertise to aid this problem, is high on demand. In this paper, our proposed system focuses on the potential risk factor for system design. Using computational technique, we prune unnecessary risk factors which are less likely to cause stroke on patient dataset collected from a medical college in Bangladesh. Fuzzy C-means classifier and Fuzzy Inference System are used to classify input data. Later on, to generate fuzzy rule we use Adaptive Neuro-fuzzy Inference System so that it can give better prediction. The developed system provides higher accuracy which satisfies the physicians' demand. Therefore, the developed system will aid not only general people but also medical experts.

Keywords: Stroke · Risk factor · Fuzzy-classifier · ANFIS · Data-mining · FCM · FIS · Bangladeshi dataset · Fuzzy rule

1 Introduction

In recent years, the number of people suffering from stroke has been reached to an alarming state throughout the world. Stroke is known as one of the five leading causes of deaths in Bangladesh [1]. According to World Health organization (WHO), the mortality rate of stroke victim is 53.59 in Bangladesh [2]. In the worst possible scenario stroke leads to the death of patient. A large number of people are living under the risk factors of this crucial disease. Early detection of stroke disease can be possible by implementing computation detecting system which includes usage of medical diagnosis information and assigning medical rules (consulted with the physicians) to detect the stroke. The field of medical informatics deals with the storage, retrieval, and optimal use of biomedical information, and knowledge for problem solving and decision-making [3]. By the collaboration of medical science and the soft computing, we can be able to achieve the desired prediction of the certain disease. The computational detecting system has the logical reasoning and the assigned medical rules of predicting disease which can minimize the human error and lack of knowledge. The objective of this research is to

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_26

develop a decision support system to reduce the chance of stroke in developing country like Bangladesh. The support system would use Fuzzy C-Means Clustering algorithm for the classification of data based on the risk factors which have gone through an initial data mining process; Fuzzy Inference System (FIS) and Adaptive Neuro-Fuzzy Interface System (ANFIS) would be used to generate fuzzy logic based rules to achieve higher accuracy on the process of early detection of stroke. One of the key features of our proposed system is the reduction of attributes by using data mining which helps in detecting main risk factors of stroke disease. The system has implemented in certain way that will give less error and cost effective fast results.

2 Literature Review

The study [4] proposed by Indira Muhic et al. designed a fuzzy approach to determine breast cancer by implementing pattern recognition and Fuzzy c-means (FCM) algorithm. The proposed system of the study showed the satisfactory accuracy using FCM algorithm with the success 100% true positive, 87% true negative, 0% false positive, 13% false negative.

The authors in [5] predicted the diabetes disease by implementing Fuzzy C Mean clustering algorithm along with the Support Vector Machines using Sequential minimal optimization to establish a comparison between the techniques. FCM gives an accuracy of 94.3% and SVM has an accuracy of 59.5%.

The authors in [6] reported accuracy of around 80.88% in the diagnosis of Parkinson Disease using Fuzzy c-means clustering (FCM) and the pattern recognition method based on the datasets of two classifications.

In this proposed study the authors in [7] aimed for the prediction of cardiovascular disease with fuzzy c-means classifier. The proposed system used the records of 270 patients and gave 13 attributes as input in FCM to classify the risk of heart attack. The classification of the clustering algorithm gives 92% accuracy.

In the study [8] the authors developed a prediction support system to predict the Thrombo-embolic stroke disease by using the Back propagation algorithm and multi layered feed forward network model of Artificial Neural Networks (ANN). The research work demonstrates that the ANN based prediction has 89% accuracy.

3 Stroke

Interruption of blood flow to brain causes deficiency of proper oxygen and nutrition to brain cells. The brain cells die due to this reason and stroke occurs. According to James McIntosh, "There are three main kinds of stroke; ischemic, hemorrhagic and transient ischemic attack (TIA)" [9]. The causes of these different types of strokes are also different. 85% of strokes are ischemic. When the arteries which provide blood to the brain narrowed down or blocked ischemic stroke occurs. Hemorrhagic stroke occurs if the arteries in the brain leak blood. TIA is caused by blood clotting, is one kind of ischemic stroke [9].

3.1 Dataset

We collect 500 Bangladeshi patient data from DMC (Dhaka Medical College). Among them 350 are diagnosed with stroke and the remaining 150 are diagnosed as negative. After consulting with a few physicians from DMC and taking patient dataset we select 16 attributes which are suitable for our local dataset and mostly responsible for stroke (Table 1).

Table 1. Selected attributes

SL no.	Attribute name
1	Age (years)
2	Sex (0-Female 1-Male)
3	Heredity (1°) (1-yes 0-No)
4	Systolic Pressure (SP) (mmHg)
5	Diastolic Pressure (DP) (mmHg)
6	Diabetes Mellitus (DM) (years-yes 0-No)
7	Total cholesterol (mg/dL)
8	Tri-glyceride (TGs) (mg/dL)
9	High-Density Cholesterol (HDL) (mg/dL)
10	Low-Density Cholesterol (LDL) (mg/dL)
11	Myocardial infraction (1-yes 0-No)
12	Stroke history (Times-yes O-No)
13	Tobacco intake (1-Yes 0-No)
14	Pain killer Intake (1-Yes 0-No)
15	Hours of physical work
16	Class (1-Stroke 0-Normal)

4 Methodology

The complete procedure of our proposed system begins with data processing. Data mining technique is applied to reduce the number of total attributes. To determine the number of clusters, cluster validation technique is applied. The classification algorithm is implemented on mined dataset. To extract rules Fuzzy Inference System and an adaptive neuro-fuzzy inference system will create. In final step, the result is analyzed by using confusion matrix. Figure 1 provides a block diagram of our system.

Data-processing: This process involves data filtration to extract missing data, noisy and over fitting data. To predict the risk of stroke, we need to convert the raw data into row-column format.

Data Format: To conduct the whole process, we should convert data files into different format, so that it can be easily reviewed by two different tools that we have used in our system. First, we convert data to.csv (comma delimited file) format which is a simple file format to represent tabular data. In this format, each line represents one sample separating each attribute by a comma. And this will be further converted into. arff file and .dat file.

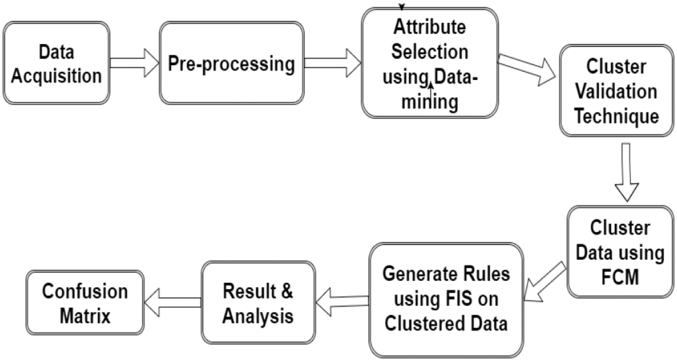

Data Acquisition
Pre-processing
Attribute Selection using Data-mining
Cluster Validation Technique
Cluster Data using FCM
Generate Rules using FIS on Clustered Data
Result & Analysis
Confusion Matrix

Fig. 1. Block diagram of proposed system

Attribute Selection: Attribute selection works by removing redundant attributes [10]. For risk level prediction of stroke, there are many risk factors suggested by medical experts. Among them, we choose a few numbers according to our observation from field work and doctor's recommendation, which best suits on our local patients. On these selected attributes, we use data mining technique to determine the most causing risk factor by mining given dataset. To minimize the number of attributes, we apply different algorithms using Weka 3.8.

To predict stroke with less number of attributes, dimensionality reduction is helpful. For any kind of medical disease it is always good if we can predict it with less number of inputs. Doctors also suggest this. For different dataset the mining result can be different and different algorithms can be appropriate like PCA (Principal Component Analysis). First, we apply PCA using ranker search method which returns 13 attributes from 16. Next, we apply best-first search and greedy approach with CfsSubsetEval method and end up with 10 attributes from 13. After applying best first search with wrapperSubsetEval and J48 decision tree it gives us 5 attributes. Finally after consulting with physicians we choose 5 final attributes which was returned by best first search algorithm, as it was the best attribute set to predict the risk of stroke on our dataset. The 5 attributes that we have used for further processing is given in Table 2.

Dunn Index (DI): For selecting cluster validation we use Dunn index. For risk level prediction Validation of the cluster analysis is extremely important because of its somewhat 'artsy' aspect [11]. Dunn index identify dense and separated clusters from many ways of defining the size or diameter of a cluster (Introduced by C. Dunn in 1974). The Dunn index is a metric for evaluating clustering algorithm. For each cluster, the Dunn index is calculated by following formula:

$$D = \frac{min_{1 \leq i < j \leq n} d(i,j)}{max_{1 < k < n} d'(k)}$$

Table 2. Final attributes

SL no.	Attribute name
1	SP (mmHg)
2	DP (mmHg)
3	Stroke history (Times-yes O-No)
4	Tobacco intake (1-Yes 0-No)
5	Pain killer intake (1-Yes 0-No)
6	Class (1-Stroke 0-Normal)

Where $d(i,j)$ is the distance between clusters I, j, and $d\ '(k)$ measures intra-cluster distance of cluster k. The algorithms are more desirable which produce clusters with high Dunn index [12]. We found c = 2 number of cluster after applying Dunn index technique in our dataset.

Fuzzy C-means Classifier: Clustering helps us to classify large number of data into smaller number of cluster or similar prototypic class. Depending on the data pattern, this algorithm classifies data into mentioned number of classes using unsupervised learning technique. In this case, algorithm was used to classify data into two clusters, as it satisfies the Dunn index validation technique. In simple word, the main purpose of clustering is to make groups from large data sets [13]. From the clustered output, we can understand clustering behavior which will lead us to generate fuzzy rules.

In this system, clustering algorithm was used to separate data into 'stroke' and 'normal' cluster depending on their input values. Developed system separate those samples from the raw data set which were not clustered as actual class. The Fuzzy clustering allows each data point to belong to more than one cluster at a time where each data point is assigned with a membership value. This membership value symbolizes that how strong one data point belongs to one cluster than the other.

We used c = 2 for our clustering algorithm. Initially the algorithm declares two cluster centers at random and measures distance for each data point from those two center and then assign a membership function for each sample. By comparing each distance with centroid algorithm continuously update membership values for each sample unless error reduction rate reach threshold. Here, in this system, the threshold is set at $1e^{-08}$ and maximum number of iterations is 100.

Figure 2 shows the flow chart of the whole FCM algorithm that have been developed in our research.

The system has used 5 attributes. To plot clustering representation, we can use 10 different combinations and from that we can also analyze rule pattern for 'stroke' and 'normal' cluster. In Fig. 3, X-axis shows SP (Systolic Pressure) values which range from 80 to 220 and Y-axis shows DP (Diastolic Pressure) values ranged from 60 to 120. The blue dots which symbolize 'normal' fall under the region where both SP and DP are low.

Fuzzy Model: In fuzzy model section, we generate a Mamdani-type fuzzy inference to dictate stroke. As we said earlier by data mining process we use just only the crucial

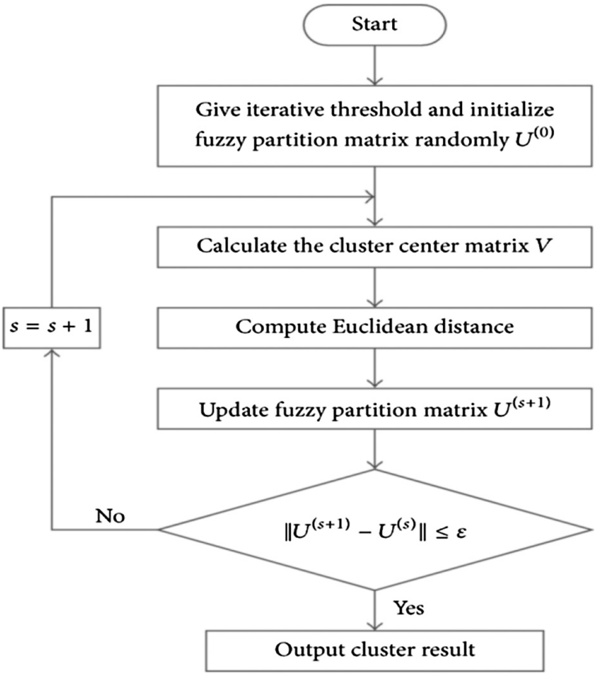

Start
Give iterative threshold and initialize fuzzy partition matrix randomly $U^{(0)}$
Calculate the cluster center matrix V
Compute Euclidean distance
$s = s + 1$
Update fuzzy partition matrix $U^{(s+1)}$
No
$\|U^{(s+1)} - U^{(s)}\| \leq \varepsilon$
Yes
Output cluster result

Fig. 2. Flow chart of FCM algorithm

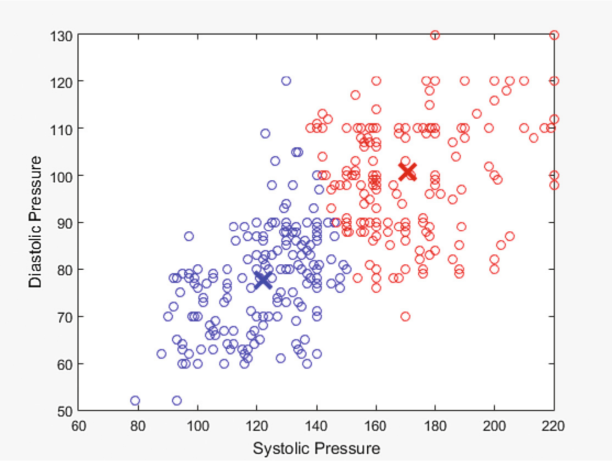


Fig. 3. Clustering result

symptoms of stroke that is most responsible for stroke. Here we do not use any irrelevant and noisy data; we remove these earlier in FCM phase.

We used 5 parameters as our input. The membership function of each input and output is two (Fig. 4). The discussion regarding membership function describe below:

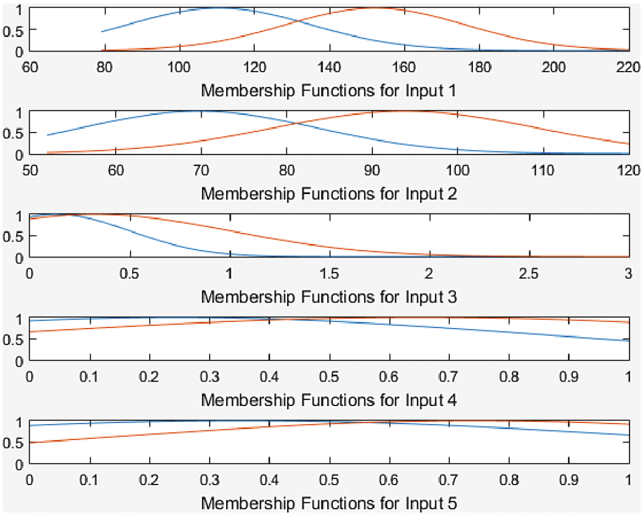


Fig. 4. Membership functions

For input-1 which is SP, the membership function for High and low is generalized as bell curve. For high value, membership function starts from 79 and ends at 220. It shows the membership value 1 when the data point is at 150. For low, membership function starts from 0 and ends at 180.

For DP, we place two fuzzy values: Low and High. The range for Low and High is given consecutively 52–120 and 0–130. Bell curve is used for membership function.

In Stroke History attribute, membership function for yes value is rising from 0 and reaches unity at 0.5 where it ends at 3. As, in our collected data set, we get stroke patient with maximum 3 pre-stroke.

In Tobacco Intake attribute, the membership function for both yes and no value are plotted over the same range. But their membership function varies in each class, which determines the membership value of each sample within the cluster.

The last and final attribute is Pain-killer intake. This attribute's membership function also follows the exact behavior of previous attribute.

Table 3 represents the modal point for each attribute where each attribute is classification range is given.

Fuzzy If-Then Rules: Fuzzy If-Then rules are very much important for predicting a disease. It gives a clear visualization. For our prediction system, we found 32 fuzzy "If-Then" rules. For example some rules are listed below:

Table 3. Modal points with FCM

SP	High 79–220	Low 0–180
DP	High 52–130	Low 0–110
Stroke history	Yes 0–3	No 0–1
Tobacco intake	Yes 0–1	No 0–1
Pain killer	Severe intake 0–1	Moderate intake 0–1

Rule 1: If (SP is Low) and (DP is Low) and (Stroke History is No) and (tobacco Intake is No) and (Pain Killer is Moderate Intake) then (Class is Normal).

Rule 2: If (SP is High) and (DP is High) and (Stroke History is Yes) and (tobacco Intake is Yes) and (Pain Killer is Severe Intake) then (Class is Stroke).

Rule 3: If (SP is Low) and (DP is High) and (Stroke History is No) and (tobacco Intake is No) and (Pain Killer is Moderate Intake) then (Class is Normal).

Rule 4: If (SP is High) and (DP is Low) and (Stroke History is Yes) and (tobacco Intake is Yes) and (Pain Killer is Severe Intake) then (Class is Stroke).

Defuzzification Method: As our system use mamdani inference, the defuzzification uses centroid method. Formula as follows:

$$\hat{t} = \frac{\int_0^{tmax} t * cd, s(t)dt}{\int_0^{tmax} t * cd, s(t)}$$

ANFIS: ANFIS is a kind of artificial neural network that is based on 'Sugeno' type fuzzy inference system. Its inference system gives a set of fuzzy IF–THEN rules [14]. We use ANFIS to optimize our fuzzy model. The approach is done in MATLAB. Our proposed ANFIS network is evaluated against training and testing dataset for classification accuracies. Our proposed ANFIS network confirms that it has potential to predict stroke (Fig. 5).

As we can see in figure, there are 5 inputs with two membership functions for each, by generating rules it gives one output.

5 Experimentation

The developed system has been implemented through the following steps:

Step 1: Collecting the desired dataset for stroke and normal people.

Step 2: Use Data mining technique to select best possible attributes on dataset.

Step 3: Use Dunn index technique to check cluster validation for all data.

Step 4: Determine c values for finding modal points by using FCM clustering algorithm.

Step 5: Separate the resultant cluster.

Step 5: Generate FIS on clustered data.

Step 6: Evaluate FIS values by calling 'evalfis' function.

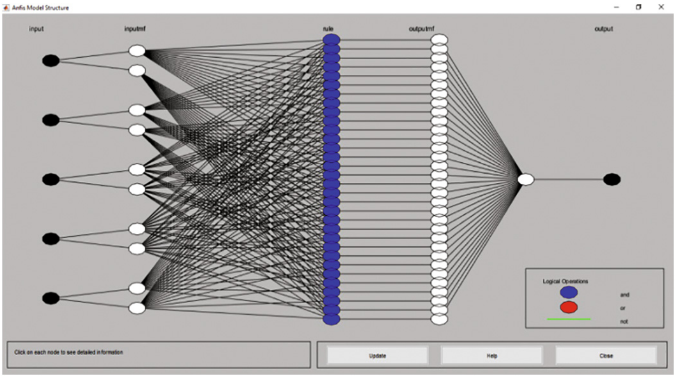


Fig. 5. ANFIS network

Step 7: Use tenfold validation technique to set testing and training data.
Step 8: Set membership function for each attribute.
Step 9: Set rules for prediction of stroke.
Step 10: Generate ANFIs network.
Step 11: Find a new set of data for stroke and normal class.
Step 12: Calculate accuracy, sensitivity and specificity by using confusion matrix.

6 Result and Analysis

In implementing phase, after taking the selective parameters the proposed system has gone through evaluation. Table 3 shows the result for test data. The total dataset divided into two set: training data and testing data. The training data contains 420 samples 84% of total data and testing-data is rest 16%. By training data the system was trained where test data is used to validate the system. Fuzzy C-means clustering is used to cluster data set. MATLAB R2016a along with fuzzy toolbox is used to build this classification algorithm. The whole data set classified into two classes. FIS is generated to give the system more accuracy by deducting the noisy data after clustering. By using 'evalfis' the accuracy was measured. Further ANFIS network is generated to validate the predicted rules. The accuracy of our system is 95.1% which is good enough to give validity to our intelligent proposed system. By using confusion matrix we evaluate the predictive model. For predicting stroke a work was done [8] in ANN where back propagation algorithm was used to predict Thrombo-embolic stroke and their accuracy was 89%. In our result, we have the accuracy is 95.1%, sensitivity is 93.78% and specificity is 100% in our overall procedure. Table 4 records those. We also record some sample output from our system in Table 5.

Table 4. Confusion matrix

		Predicted class	
		Yes	No
Actual class	Yes	226(TP)	15(FN)
	No	0(FP)	126(TN)

Table 5. First 10 samples from datasets with FIS results

SP	DP	Stoke history	Tobacco intake	Pain-killer intake	Class	FIS
170	80	0	1	1	1	1
160	100	1	0	1	1	1
140	80	0	1	1	0	0
220	100	0	1	1	1	0
110	80	0	1	1	0	0
160	100	0	1	1	1	1
150	80	2	1	1	1	1
188	80	1	1	1	1	1
140	100	0	1	0	0	0
160	100	0	1	0	1	1

7 Conclusion

In this project, we worked on Data mining, FCM clustering, FIS and ANFIS particularly for Bangladeshi dataset so that we can generate more accurate rules. We found 95.1% accuracy for predicting stroke. The implemented system is easily understandable by others. As a future work, we plan to implement prediction system for each type of stroke particularly. In our future work it will be our main concern to implement a way which will be more accurate and more efficient by considering more patients from different hospitals.

Acknowledgment. The authors of this project would like to thank North South University, to give an opportunity to work here. We also thank Dhaka Medical College for allowing us to conduct the survey and Dr. Yeasir Arefin Ovi, a practicing Doctor of Dhaka Medical College for supporting us throughout our project. We would like to convey our gratitude to Dr. Kapil Rahman, ICU duty doctor of Holy Family Medical College, Dhaka, Bangladesh for guiding us with his medical knowledge and findings.

References

1. CDC, CDC global health - Bangladesh, CDC (2014). https://www.cdc.gov/globalhealth/countries/bangladesh/. Accessed 7 Jan 2017
2. Islam, N., et al.: Burden of stroke in Bangladesh. Int. J. Stroke **8**(3), 1–3 (2012). https://www.researchgate.net/publication/230847528_Burden_of_stroke_in_Bangladesh. Accessed 7 Jan 2017
3. Bernstam, E.V., Smith, J.W., Johnson, R.: What is biomedical informatics? J Biomed Inform. **43**(1) (2010). https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2814957/. Accessed 7 Jan 2017
4. Muhic, I.: Fuzzy analysis of breast cancer disease using Fuzzy C-Means and pattern recognition. SE. Eur. J. Soft Comput., 50–55 (2009). http://scjournal.ius.edu.ba/index.php/scjournal/article/viewFile/45/45. Accessed 7 Jan 2017
5. Sanakal, R., Jayakumari, S.: Prognosis of diabetes using data mining approach-fuzzy C Means clustering and support vector machine. Int. J. Comput. Trends Technol. **11**(2), 94–98 (2014)
6. Rustempasic, I., Can, M.: Diagnosis of Parkinson's disease using Fuzzy C-Means clustering and pattern recognition. SE. Eur. J. Soft Comput. **2**, 42–49 (2013)
7. Chitra, R.: Heart attack prediction system using Fuzzy C Means classifier. IOSR J. Comput. Eng. **14**(2), 23–31 (2013)
8. Dhanushkodi, S., Sahoo, G., Nallaperumal, S.: Designing an artificial neural network model for the prediction of Thrombo-embolic stroke. Int. J. Biometrics Bioinform. (IJBB), **3**(1) (2009). http://www.cscjournals.org/library/manuscriptinfo.php?mc=IJBB-7. Accessed 7 Jan 2017
9. McIntosh, J., Webberley, H.: Stroke: Causes, symptoms, diagnosis and treatment. Medical News Today (2016). http://www.medicalnewstoday.com/articles/7624.php. Accessed 7 Jan 2017
10. Witten, I.H.: More Data Mining with Weka
11. Cluster validation. https://hlab.stanford.edu/brian/cluster_validation.html. Accessed 7 Jan 2017
12. Cluster analysis, in Wikipedia, Wikimedia Foundation (2017). https://en.wikipedia.org/wiki/Cluster_analysis. Accessed 7 Jan 2017
13. MathWorks, T.: Data clustering (1994). https://www.mathworks.com/help/fuzzy/data-clustering.html?requestedDomain=www.mathworks.com. Accessed 7 Jan 2017
14. Adaptive neuro fuzzy inference system, in Wikipedia, Wikimedia Foundation (2016). https://en.wikipedia.org/wiki/Adaptive_neuro_fuzzy_inference_system. Accessed 7 Jan 2017

Weighted Similarity: A New Similarity Measure for Document Ranking Features

Mehrnoush Barani Shirzad[1(✉)] and Mohammad Reza Keyvanpour[2]

[1] Data Mining Research Laboratory, Department of Computer Engineering,
Alzahra University, Tehran, Iran
mb.shirzad@yahoo.com

[2] Department of Computer Engineering, Alzahra University, Tehran, Iran
keyvanpour@alzahra.ac.ir

Abstract. Many ranking features are utilized by information systems. Several ranking methods act similarly to each other and thus provide similar information. Some information retrieval systems need to select privilege ranking methods and eliminate redundant rankers. To deal with redundant features, the present work introduces a new feature similarity measure, which is based on documents distance. Then the measure is weighted by relevance degree of documents. Experiments are conducted on two data sets MQ2008 and OHSUMED for all features pairs. We adopt two methods of similarity measures in order to compare them with our similarity measure. Results show that our method has correlation with other measures and with MAP.

Keywords: Information retrieval · Similarity measure · Ranking similarity

1 Introduction

Ranking the results is a challenge in every information retrieval system. As an example of ranking, ranker algorithms provide a list of documents according to their relevance. Many algorithms have been developed for document ranking, including vector space algorithms and probabilistic algorithms. The rankers are compared with each other according to their results with the aim of deducing their similarity. Evaluating the ranked list is conventionally performed by evaluation measures including NDCG, MAP, ERR and can be applied for comparison. Similarity measures show how much a pair of rankers are correlated. A similarity measure indicates the agreement of two rankers with a value which is often in range of [0..1] or [−1..1]. This similarity value can apply under feature selection task in systems that apply a variety of rankers to compare the results of two search engines. In this paper the first task is considered.

Various similarity methods have been introduced. Kendal's τ, Pearson correlation coefficient, Spearman rank correlation coefficient are well-known correlation coefficient measures. Kendal's τ considers priority between each two objects and Spearman shows the linear correlation coefficient for two ranked lists. Studies in [1–8] mentioned problems with Kendal's τ and attempted to solve the problems. Yilmaz et al. in [1] view the problem of Kendal's τ as ignoring the importance of top ranks. In [3] the problem with Spearman's footrule and is discussed and the relevance of elements and

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_27

positional information are taken into consideration. In [1] a correlation measure was applied to determine the similarity between two search engines. We investigate the issue of feature similarity, which can be applied for feature selection.

In this paper, we propose a similarity method to measure the agreement between two ranked lists based on the difference in the position of documents on each sorted list. Moreover, we consider the difference between relevance degrees of documents in each position. The experiments were conducted on two standard datasets from Letor3 and Letor4 for all features pairs. Our method contributes to correlated performance with respect to other similarity measures.

This paper is organized as follows: in Sect. 2 related work focusing on similarity measures are reviewed. In Sect. 3 we present our method, introducing the weighted similarity measure, and in Sect. 4 empirical results are reported, and we conclude in Sect. 5.

2 Related Work

In this section, common rank correlation coefficient metric and the algorithms introduced to improve them are reviewed.

The aim of a similarity measure is to evaluate the correlation of two features that rank a list of documents. Several proposals have been made for rank correlation. Kendal's τ correlation coefficient and Pearson correlation coefficient are two widely used correlation coefficient measures that are also applied for feature selection for learning to rank.

Kendal's τ [9] indicates the number of paired documents of the same list sorted by two rankers that take equal preferences order in two ranked lists, over total number of documents pairs. We apply a version of Kendal's τ as follows. Kendal's -τ (TAU) for ranking a query q and two features f_i and f_j, is defined as follows:

$$Tau\left(f_i, f_j\right) = \frac{\#\left\{(x_s, x_t) \in X_q | x_s <_{x_i} x_t \; and \; x_s <_{x_j} x_t\right\}}{\#\left\{(x_s, x_t) \in X_q\right\}} \tag{1}$$

Where the numerator is the number of paired documents related to query q that takes equal preference according to two features fi and fj, and the denominator is the number of whole pairs of documents associated with the query.

To measure the similarity, Pearson (PCC) is defined as follows:

$$PCC\left(f_i, f_j\right) = \frac{\operatorname{cov}\left(f_i, f_j\right)}{\sqrt{\operatorname{var}(f_i).\operatorname{var}\left(f_j\right)}} \tag{2}$$

Where $cov\left(f_i, f_j\right) = \sum_{k=1}^{n}\left(f_i^{(k)} - \overline{f_i}\right)\left(f_j^{(k)} - \overline{f_j}\right)$ is the covariance of two features and $var(f_i) = \sum_{k=1}^{n}\left(f_i^{(k)} - \overline{f_i}\right)^2$ is the variance of a feature.

Algorithms in [1–3] investigate the problem with Kendal's tau in ranking. In [1] Yilmaz et al. the weakness of Kendal's tau in mirroring the error and precision in top rank is discussed. They proposed τ_{AP} a variation of Kendal's tau based on the average precision that gives more weight to the errors at high rankings. τ_{AP} has gained a considerable interest [2, 3], Stefani et al. [2] apply a τ_{AP} in Mallows model as the distribution function for the problem of learning probabilistic models for permutations. Urabno et al. in [3] also apply statistical estimators Kendal's and τ_{AP} to estimate the expected correlation of test collection against true ranking.

Carterette et al. in [4] solves the problem using Kendall's tau that works regardless of actual correlation between the measurements, proposing a rank correlation based distance between rankings. In [5] Kumar et al. view the problem with Spearman's footrule and Kendall's tau as overlooking objects relevance and positional information. They extend Spearman's footrule and Kendall's tau to element weights, position weights, and element similarities. They list five principles for similarity metric; richness, simplicity, generalization, basic properties and correlation with other metrics. Richness evolves three concepts: element weights, position of elements and diversity. In [6] Luchen et al. propose a family of similarity measures based on maximization effectiveness difference. Effectiveness measures include MAP, NDCG and ERR applied. Webber et al. in [7] address the problem of indefinite rankings in which two lists lack the same items and only have some common items. Gao et al. in [8] by considering the top of the ranked list, propose a head-weighted measure. Their metric evaluates the gap between system scores, and is effected by the gap at the top of the ranked lists. In the next section, our method is presented based on distance and relevance degree.

3 Our Method

We consider feature (ranker) similarity under feature selection problem.

3.1 Similarity Based on Distances

A similarity measure is applied to evaluate the different between ranker's results. To evaluate the similarity between features (ranking method) according to their ranked list, one way is to consider the distance between two ranked lists. We also consider the relevance (importance) degree of each document on the ranked list. Figure 1 shows the weighted similarity measure pattern.

We apply the definition of distance from Spearman's ρ, and present a new similarity measure. The distance of instances in two ranked lists of the same document, provided by two features (ranker) is defined as the difference in the two positions on the two ranked lists. The following demonstrates an example for the distance of documents in two lists ranked on the same set with two different features.

Feature f_i: $d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8, d_9$
Feature f_j: $d_9, d_8, d_5, d_7, d_6, d_1, d_3, d_4, d_2$
Distance of d_3 = position d_3 in list 1 - list 2 (|3–7|)

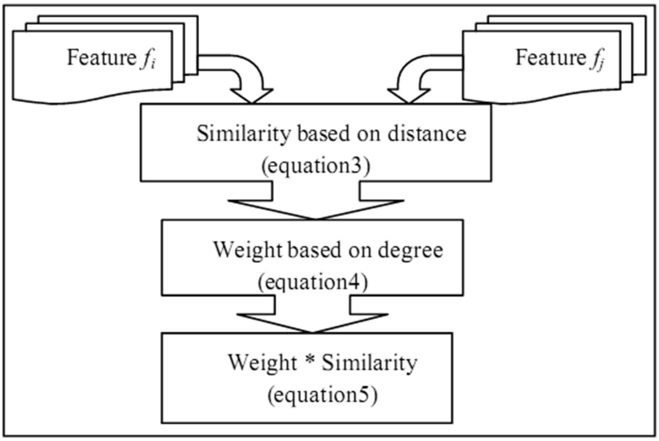


Fig. 1. Weighted similarity measure pattern

To define similarity the following normalization is used:

$$sim\left(f_i, f_j\right) = \frac{\max(d) - \sum d}{\max(d)} \tag{3}$$

Where max (d) is the maximum distance which, for n instances, equals $\left\lfloor \frac{n^2}{2} \right\rfloor$. The above definition measures the similarity according to difference in distance, but the relevance of each document in each position is important in ranking.

3.2 Similarity Based on Degree

We consider another distance, which shows the difference of weight in two ranked lists. For two ranked lists the relevance of each document in each position is considered.

For each position the absolute difference of relevance degrees of two ranked lists is computed. This distance is based on the degree of relevance. The following example illustrates the distance of degrees.

docs degree:	$d_1 = 0$, $d_2 = 1$, $d_3 = 2$, $d_4 = 0$, $d_5 = 2$, $d_6 = 1$, $d_7 = 3$, $d_8 = 0$
Feature f_i:	d_3, d_2, d_3, d_4, d_7, d_5, d_6, d_8
Feature f_j:	d_1, d_2, d_3, d_8, d_5, d_6, d_7, d_4
Distance of degrees:	2, 0, 0, 0, 1, 2, 0

Sum of this distance shows the weighting difference of two ranked lists. For two identical rankings, this value is zero and the maximum value is (r.$\bar{\text{d}}$) in which r is the number of degrees and d is the total object or position. Then d is normalized to the following equation to define similarity:

$$w(f_i,f_j) = \frac{\max(\overline{d}) - \sum \overline{d}}{\max(\overline{d})} + \alpha \tag{4}$$

Where α is a threshold to prevent zero, which equals 1 for identical rank. The similarity measure becomes w.sim.

$$w.sim(f_i,f_j) = \left(\frac{\max(\overline{d}) - \sum \overline{d}}{\max(\overline{d})} + \alpha \right) . \frac{\max(d) - \sum d}{\max(d)} \tag{5}$$

4 Experiments

In this session, first the datasets and evaluation measures are introduced and then the algorithm settings are presented.

4.1 Dataset

In order to investigate the proposed method we conduct our experiment on two datasets from Letor benchmark. MQ2008 from Letor 4.0 has in total 784 queries, containing 15211 document-query pairs, for which 3 relevance degrees have been provided and 46 features have been extracted. OHSUMED dataset from Letor 3.0 consists of 106 queries, 45 features extracted for each document query pair. It consists of 16 140 document-query pairs, and 3 relevance degrees are supplied. For the purpose of cross-validation, each dataset is folded into five folds and each fold contains a training set, a validation set and a test set.

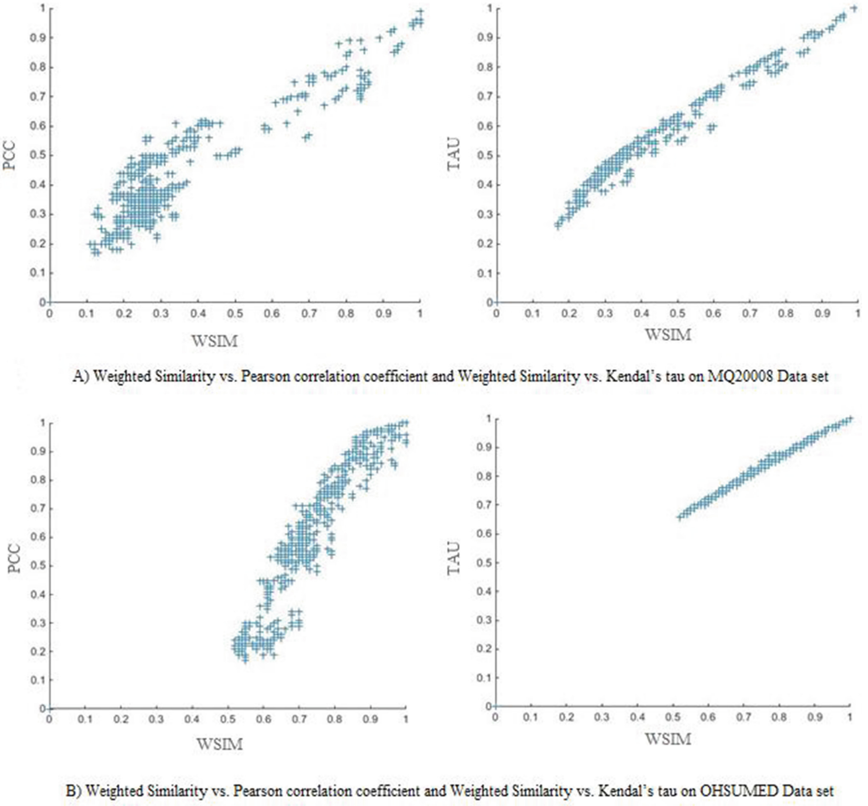
To compare the proposed method with other correlation strategies we applied two similarity methods, Kendal's -τ Eq. (1) and Pearson correlation coefficient Eq. (2).

To assess the proposed measure and show that similar features provide similar results according to the accuracy measure, a comparison between the results of weighted similarity against MAP is necessary. The problem is the correlation between two ranked lists, none of which is the ground truth. Therefore, we compare weighted similarity against their similarity in MAP. For this evaluation we define MAP similarity of two features as MAPSIM $(f_i,f_j) = 1 - |\text{MAP}(f_i) - \text{MAP}(f_j)|$.

In the following WSIM shows the weighted similarity, PCC is used to refer to Pearson correlation coefficient and TAU shows Kendal's tau. MAP represents MAPSIM.

4.2 Experimental Results

In Fig. 2(A) the plot illustrates the weighted similarity against Pearson correlation coefficient and Kendal's tau across all pairs of features for MQ2008. The plot shows a correlation between weighted similarity against Kendal's tau. Almost all data follow a similar pattern. The weighted similarity is partly correlated with Pearson correlation coefficient.



A) Weighted Similarity vs. Pearson correlation coefficient and Weighted Similarity vs. Kendal's tau on MQ20008 Data set

B) Weighted Similarity vs. Pearson correlation coefficient and Weighted Similarity vs. Kendal's tau on OHSUMED Data set

Fig. 2. Weighted similarity vs. Pearson correlation coefficient and weighted similarity vs. Kendal's tau

In Fig. 2(B) the plot shows the results for OHSUMED containing weighted similarity against Kendal's tau and Pearson correlation coefficient across all pairs of features. The plot shows again that for this data set the correlation between weighted similarity and Kendal's tau is significant. The weighted similarity and Pearson correlation coefficient are partly correlated.

Figure 3 plots the similarity between MAP of pairs of features against weighted similarity of corresponding feature pairs. According to plot A, there is a correlation between MAP and weighted similarity for OHSUMED data set. However, MAP similarity produced a higher value compared with weighted similarity. The linear relation between two measures for all pairs of features is visible, which shows similar features based on the proposed measure providing a similar accuracy according to MAP. A correlation between MAP and weighted similarity for MQ2008 data set is demonstrated in plot B. Though this correlation is linear and less gradual than the plot for OHSUMED, the data set shows that there is a clear relation between the two measures.

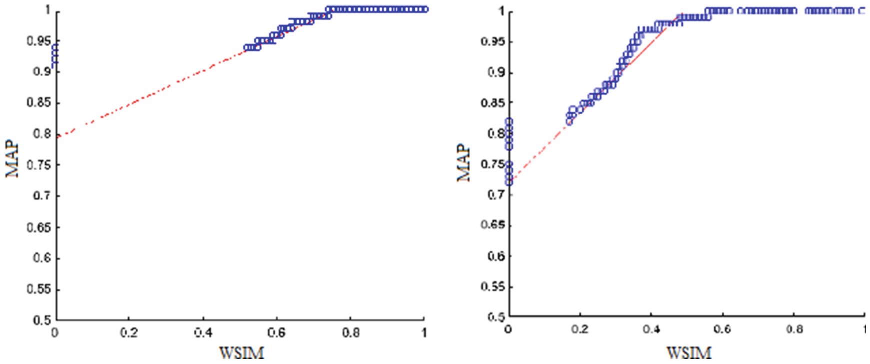


Fig. 3. (A) Weighted similarity vs. MAP similarity on OHSUMED Data set (B) Weighted Similarity vs. MAP Similarity on MQ2008 Data set

5 Conclusion

We introduced a new similarity measure that evaluates the rank correlation between two ranking features. We applied two methods in order to compare the proposed method as similarity measures. The empirical results showed that our method is correlated with other methods and with MAP. Also the proposed measure has other properties including simplicity and is weighted based on document relevance. We conducted our experiments on document retrieval; future work will include applying weighted similarity for other information retrieval applications.

References

1. Yilmaz, E., Aslam, J.A., Robertson, S.: A new rank correlation coefficient for information retrieval. In: Proceedings of the 31st International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 587–594. ACM (2008)
2. Stefani, L.D., Epasto, A., Upfal, E., Vandin, F.: Reconstructing hidden permutations using the average-precision (AP) correlation statistic. In: Proceedings of the 13th AAAI Conference on Artificial Intelligence, pp. 1526–1532. AAAI Press (2016)
3. Urbano, J., Marrero, M.: toward estimating the rank correlation between the test collection results and the true system performance. In: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1033–1036. ACM (2016)
4. Carterette, B.: On rank correlation and the distance between rankings. In: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 436–443. ACM (2009)
5. Kumar, R., Vassilvitskii, S.: Generalized distances between rankings, In: Proceedings of the 19th International Conference on World Wide Web, pp. 571–580 (2010)

6. Tan, L., Clarke, C.L.A.: A family of Rank similarity measures based on maximized effectiveness difference. IEEE Trans. Know. Data Eng. **27**, 2865–2877 (2014)
7. Webber, W., Moffat, A., Zobel, J.: A similarity measure for indefinite rankings. ACM Trans. Inf. Syst. **28**(4), 20 (2010)
8. Gao, N., Bagdouri, M., Oard, D.W.: Pearson rank: a head-weighted gap-sensitive score-based correlation coefficient. In: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 941–944. ACM (2016)
9. Kendall, M.: Rank Correlation Methods. Oxford University Press, Oxford (1962)

Face Identification and Face Classification Using Computer Vision and Machine Learning Algorithm

Suhas Athani(✉) and C.H. Tejeshwar

B.V. Bhoomaraddi College of Engineering and Technology, Hubli, India
suhas2012athani@gmail.com, chtejeshwar4@gmail.com

Abstract. This paper aims to address a mechanism to recognize and classify the faces present in the videos. The number of videos are collected from the different sources. Each of the videos are divided into frames which are images. Key frames are extracted and stored in a separate directory. From the extracted key frames, faces present in them are detected. These set of faces detected will be used as a training data-set which are stored in normal directory. The features from each of the face images are extracted by making use of Local Binary Patterns (LBP). The feature values will further be used to train the Support Vector Machine (SVM) classifier. In the testing phase, the same process is carried out and SVM makes the prediction based on the training data-set to which class the face belongs to. The effectiveness of the proposed system's prediction is demonstrated by collecting different video data-set and the accuracy of the model is comparable.

The proposed system aims to classify two persons and accuracy of the model is calculated using K-fold cross validation.

Keywords: Key frame extraction · Local binary patterns (LBP) · Support vector machine (SVM)

1 Introduction

In this paper, face of the person present in a video is identified and further prediction is made whether the face of a person belongs to class-1 or class-2 with the help of linear based support vector machine (SVM) classifier. We are classifying two persons; Class-1 means a face of first person while class-2 means the second person. A single person can exhibit many facial expressions like happiness, sadness and so classification of faces with different facial expressions, illuminations in a video is a challenging problem. To address this topic, the video is divided into frames. Key frames are extracted. If a face is present in an extracted key frame it is identified and features are extracted from the respective key frames. The feature extracted values is used to build a model for prediction of the class labels (class-1 or class-2) using SVM. The classification of faces into different classes find its applications in the areas of content based retrieval, face detection, face recognition and face tracking [1].

In [2], authors propose a new pattern classification method called Nearest Feature Line (NFL), which has been shown to yield good results in face recognition and audio

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_28

classification and retrieval. In [3], author extended the NFL method to video retrieval. Unlike conventional methods such as NN and NC, the NFL method takes into consideration of temporal variations and correlations between key-frames in a shot. The main idea is to use the lines passing through the consecutive feature points in the feature space to approximate the trajectory of feature points. In [4], the idea of Eigen face is introduced, which is one of earliest successes in the face recognition research and successfully applied the texture descriptor, local binary pattern (LBP), on the face recognition problem. In [5], author proposes the use of sparse representation derived from training images for face recognition. The method is proved to be robust against occlusions for face recognition.

In [6], some researches also use a reference set to improve the accuracy of face recognition and retrieval and used attribute classifiers, SVM classifiers trained on reference set, for face verification. Methods for off-line recognition of hand printed characters have successfully tackled the problem of intra-class variation due to differing writing styles. However, such approaches typically consider only a limited number of appearance classes, not dealing with variations in foreground/background color and texture [7].

The rest of the paper is organized as follows. Section 2 describes proposed approach. The sequential approach of key frame extraction is discussed in Sect. 3. The statistical feature extraction using LBP are detailed in Sect. 4. Classification using SVM is explained in Sect. 5. The results are mentioned on the considered data-set in Sect. 6 and K-fold cross validation is used to obtain the accuracy of the model. Finally we conclude in Sect. 7.

2 Proposed System

The proposed system Fig. 1, consists of three main modules. They are key frame extraction, feature extraction using LBP, classification using SVM classifier. The system also involves two phases, training and testing phase. In the training phase, training videos are divided into frames. Only the key frames are extracted for which sequential key frame technique is used. These key frames (images) are stored in a separate directory. Furthermore, faces present in the images are detected from which features are extracted using LBP and for the convenience histogram is plotted to store the count of each LBP value. In the next phase, a model is built using SVM classifier using training data-set. During training phase, labels are assigned to the classes considered. For class-1, label provided is 1 and for class-2, label provided is 0. In testing phase, separate testing data-set is considered which are unlabeled. In this phase, SVM classifier makes a prediction whether the face belongs to class-1 or class-2. Class-1 is named as face1 and class-2 is named as face2. Thus, linear SVM is used to classify two persons.

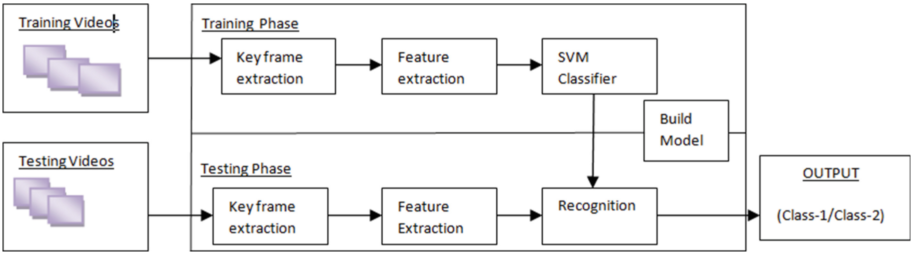


Fig. 1. System model showing the steps involved in proposed system

3 Key Frame Extraction

Video contains massive quantity of information at different levels in terms of scenes, shots and frames. The objective here addressed is the removal of redundant data which makes further processing easier. So, key frame extraction is the fundamental step in any of the video retrieval applications. Key frames are the frames which provide the summarized information of the complete video. Key frames are selected based on their uniqueness when compared to their subsequent frames. Dissimilarity between the frames must be computed in order to detect the key frames. The proposed system makes use of sequential comparison method wherein the first extracted key frame is compared with all other frames. This process is carried out until a different key frame is obtained. The sequential key frame extraction method is easy to implement as it has low computational complexity [8, 9].

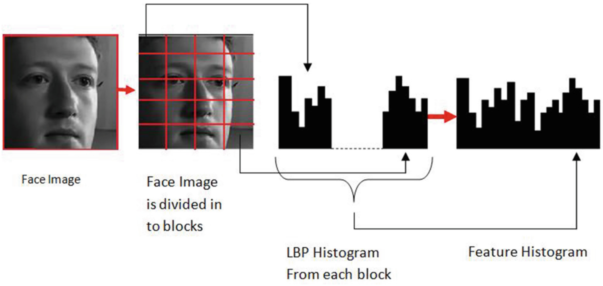
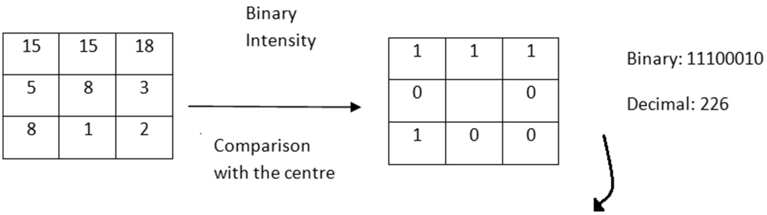

Face Image
Face Image is divided in to blocks
LBP Histogram From each block
Feature Histogram

Fig. 2. Phases involved in Linear Binary Pattern

4 Feature Extraction Using LBP

Local Binary Pattern (LBP) is widely used in the field of computer vision as a type of optical identity. This algorithm is helpful and serves as a powerful feature for texture classification. Steps carried out in LBP:

1. LBP looks at 9 pixels at a time.
2. It looks at 3 × 3 pixels and particularly interested at the central pixel.
3. For example, if the central pixel is 8 as shown in Fig. 3.



15	15	18
5	8	3
8	1	2

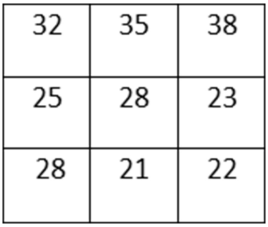
1	1	1
0		0
1	0	0

Fig. 3. Conversion of 1-byte of data into decimal carried during LBP computation

4. The central pixel is compared with the neighboring 8 pixels. If the neighboring pixel value is greater than or equal to central pixel value, then we assign 1 or else 0.
5. The binary values from the Fig. 3 are noted down as 11100010. This binary value will be converted into a decimal number which will be used to train the system. Here the binary values are noted in a clockwise manner [10] (Fig. 2).

The main advantage about LBP is that it is illumination invariant. If the lighting in the image is increased, the pixel values will also rise but the relative difference between the pixels will remain the same.

Consider the Fig. 4, 32 is greater than 28, so LBP value remains the same irrespective of illumination variation.



32	35	38
25	28	23
28	21	22

Fig. 4. Example for illumination invariant behavior of LBP

Consider Fig. 5, it is an image which is divided into 9 blocks. LBP also helps to detect the edges in a face like outline of mouth, eyelids. In the Fig. 5, three 1's and next 0, this transition indicates there are edges. By this, it is easy to make out the dark areas and light areas in the face. So basically there is a conversion from high dimensional space into a low dimensional space that only encodes relative intensity values and in doing so encodes edges.

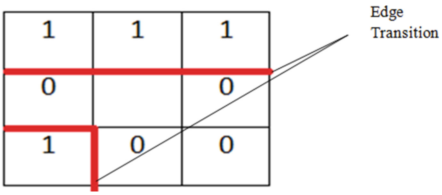


Fig. 5. Example to show transition of values helps to detect edges in LBP

5 Classification Using SVM

Classification is a process which involves separation of classes based on extracted features. Classes formed will be different from each other. The classifier is built by considering regularity patterns of training data-set. There are different classification algorithms like k-nearest neighbors, decision tree learning and support vector machine which are majorly used in various types of classification. K-nearest neighbor algorithm (kNN) uses the k-nearest neighbors to build the selection of class assignment straight from the training example. There are many algorithms like C4.5 to construct decision trees to predict the class of the input. SVM uses the concept of hyperplanes to predict the class of the input [11]. In the proposed system, SVM is used for classification of two persons.

Support Vector Machine (SVM) makes use of hyperplane that acts as boundary which divides two classes. The position of the hyperplane has to be decided for better classification. In the Fig. 6, the circles represent the features belonging to class C_1 (class-1) and triangles represent the features belonging to class C_2 (class-2). The position of hyperplane as shown in the Fig. 6, is not desirable because it gives a large bias in favor of class C_2 whereas it puts penalty with class C_1.

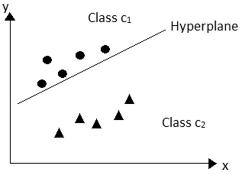


Fig. 6. Undesirable positioning of hyperplane

The reason is that, an interspace which is represented as black region in the Fig. 7, is given to the class C_2 and margin for class C_1 is less.

The classifier provides more appropriate results if the position of hyperplane is as shown in the Fig. 8, which is at an equal distance from the two classes. For Support Vector Machine, 1 (class-1) and 0 (class-2) are assigned as labels for two classes [12]. The circles and triangles that lie on the line as shown in Fig. 8 are support vectors. Decision function for a Support Vector Machine (SVM) is completely identified by

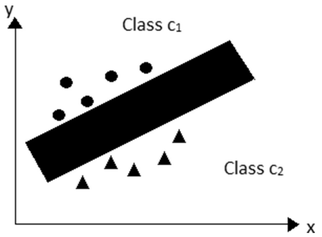


Fig. 7. Undesirable positioning of hyperplane

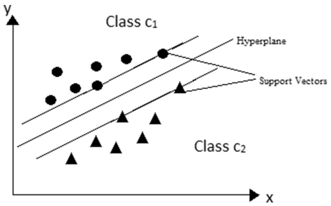


Fig. 8. Desirable positioning of hyperplane

batch of information which defines the location of hyperplane. Support Vector Machine (SVM), specifically explains the norm about selection boundary which is best, when it is distant off from whichever data instant. Presently, the classes are class-1 and class-2, and the features are the statistical texture features extracted for each image. Basically the SVM's use hyperplanes to separate different classes through an ideal function committed from training data as follows:

$$g(x) = ax + b \tag{1}$$

$ax + b = 0$, represents a hyperplane in multi-dimensionality

a – vector perpendicular to the hyperplane
b – position of the hyperplane in the d-dimensional space

For every feature vector X, the linear function $ax + b$ has to be computed.

Considering two classes C_1 and C_2, a feature vector x_1, if function lies on the positive side of the hyperplane then,

$$g(x_1) = ax_1 + b > 0, \quad x_1 € C_1 \tag{2}$$

If function lies on the negative side of the hyperplane then,

$$g(x_1) = ax_1 + b < 0, \quad x_1 € C_2 \tag{3}$$

and if function lies on the hyperplane then,

$$g(x_1) = ax_1 + b = 0 \tag{4}$$

Classifier has to be trained initially i.e., to find a and b. Supervised learning is used for training. The calculated a and b values are taken and for every sample $ax + b$ is checked. If a sample from class C_1 is chosen and if the value of $ax + b$ is not greater than zero, then the values of a and b are modified in such a way that the position of hyperplane is so modified that particular x taken from C_1 is moved to positive side of the hyperplane.

The aim of SVM takes care of maintaining maximal distance between the feature vectors of two separating classes [13]. For every x_i, we can have y_i which represents the belongingness (± 1). Therefore, $y_i(ax_i + b)$ is always greater than zero irrespective of the class. Now $ax_i + b = \gamma$ where γ is the margin which is the measure of distance of x_i from the separating plane. Considering $ax + b = 0$, the distance of a point x from the hyperplane is given by,

$$\frac{ax + b}{||a||} \geq \gamma \tag{5}$$

where, $||a||$ tells the orientation of the plane.

$$ax + b \geq \gamma ||a|| \tag{6}$$

such that $\gamma||\mathrm{a}|| = 1$.

hence,

$$ax + b \geq 1 \; if \; x \epsilon C_1 \tag{7}$$

$$ax + b \geq 1 \; if \; x \epsilon C_2 \tag{8}$$

we have,

$$y_i(ax_i + b) \geq 1 \tag{9}$$

therefore, we can conclude that,

if $y_i(ax_i + b) > 1$, then $\mathrm{x_i}$ is not a support vector. And, if $y_i(ax_i + b) = 1$, then x_i is a support vector.

SVM is a linear machine whose design is greatly influenced by the position of support vectors. The distance of the point x_i from the plane has to be maximized.

From Eq. 5, $(ax + b)$ should be maximized and $||a||$ should be minimized.

From Eq. 9, it can be observed that $y_i(ax_i + b) \geq 1$ acts as a constraint. This constraint problem can be converted to un-constraint one by using Lagrange's multiplier.

we have,

$$L(a, b) = \frac{1}{2} \tag{10}$$

6 Results and Discussion

A set of 100 images of both class-1 and class-2 was taken from the key frames. The collected images were then cropped manually to be of same size. The data-set has 40 class-1 images and 40 class-2 images which are named as c1 to c80. The remaining 10 images are used for testing. All the images classified as class-1 are named as face1 and images classified as class-2 are named as face2. They are named likely because images named as face1 is the face of one person whose face is different from the images named as face2.

The Fig. 9, gives training image data-set which has been utilized to build SVM model and also depicts which all images are of class-1 and class-2 (Fig. 10).

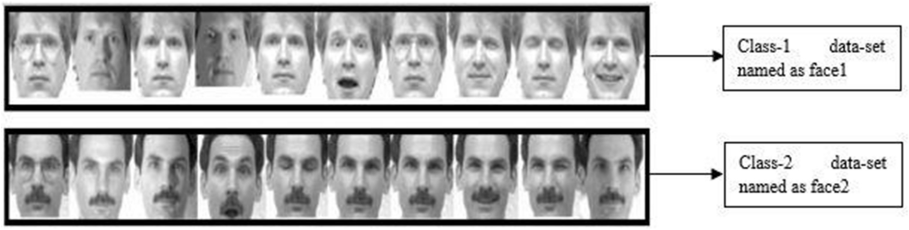

Class-1 data-set named as face1
Class-2 data-set named as face2

Fig. 9. Training data-set

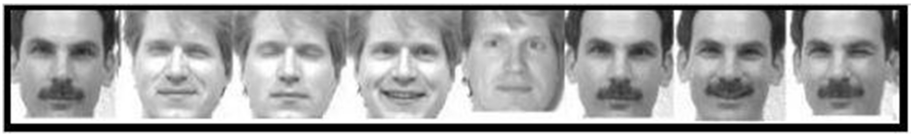
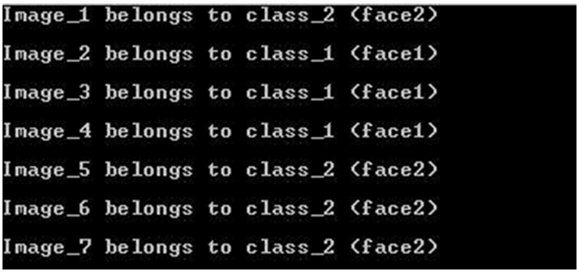


Fig. 10. Testing data-set

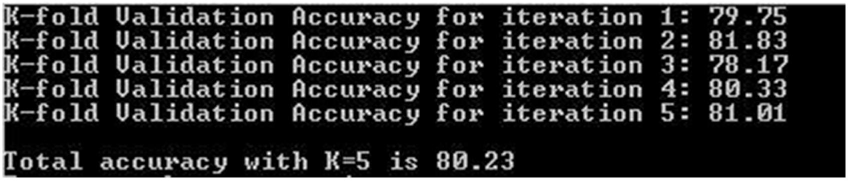
The Fig. 11, is the output which shows the prediction made by the SVM classifier. The output contains the image name and it's classification as belonging to class-1 (face1) or class-2 (face2).



```
Image_1 belongs to class_2 (face2)
Image_2 belongs to class_1 (face1)
Image_3 belongs to class_1 (face1)
Image_4 belongs to class_1 (face1)
Image_5 belongs to class_2 (face2)
Image_6 belongs to class_2 (face2)
Image_7 belongs to class_2 (face2)
```

Fig. 11. Results of image classification using SVM classifier

The classifier built can be evaluated by evaluation technique such as k-fold cross validation. Cross-validation is a model validation technique for estimating the performance of an independent data-set. In K-fold cross validation, entire data-set is partitioned into initial 80 images for the training phase and rest 20 images for the testing phase. Likewise, the process is continued for complete data-set. Finally, the exactness is studied by computing mean of 5 repetitions. During the process of K-fold cross-validation, number of folds considered are 5. The accuracy is estimated to be 80.23%. The pseudo code is as follows:



```
K-fold Validation Accuracy for iteration 1: 79.75
K-fold Validation Accuracy for iteration 2: 81.83
K-fold Validation Accuracy for iteration 3: 78.17
K-fold Validation Accuracy for iteration 4: 80.33
K-fold Validation Accuracy for iteration 5: 81.01

Total accuracy with K=5 is 80.23
```

Fig. 12. Results of K-fold cross validation

The Fig. 12, describes the result of the same as method in the Algorithm 1, where the accuracy of classification for the first, second, third, fourth and fifth iteration are namely, 79.75%, 81.83%, 78.17%, 80.33% and 81.01%. The complete accuracy is calculated as follows:

$$Total\,accuracy = \frac{Accuracy\,of\,(iteration\,1 + iteration\,2 + iteration\,3 + iteration\,4, + iteration\,5)}{5} \tag{11}$$

```
divide train data into 5 parts
for i = 1 to 5
train SVM using 80 images
compute accuracy using 20 images
end for
compute average accuracy of the 5 runs and express it in terms of percent
```

Algorithm 1: Usage of K-fold Cross Validation in Calculating Accuracy of the model

7 Conclusion

In this paper, faces of humans are detected and classified using LBP and SVM algorithms. The key frames are extracted from the videos. From these key frames, faces are identified and features are extracted using LBP. The classification of faces is carried out using linear SVM classifier. Further, work will be considered for more than two classes, where the implementation and processing can be carried out in a parallel and distributed manner. This is required because the time requirement for SVM increases with an increase in the number of classes.

References

1. Li, S.Z., Jain, A.K.: Handbook of Face Recognition. Springer, London (2005)
2. Li, S.Z., Lu, J.: Face recognition using the nearest feature line method. IEEE Trans. Neural Netw. **10**(2), 439–443 (1999)
3. Zhano, L., Qi, W., Li, S.Z., Yang, S.-Q., Zhang, H.J.: Key-frame extraction and shot retrieval using nearest feature line (NFL). Technical report, China (2000)
4. Chen, B.-C., Chen, C.-S., Hsu, W.H.: Face recognition and retrieval using cross-age reference coding with cross-age celebrity dataset. IEEE Trans. Multimed. **17**(6), 804–815 (2015)
5. Naik, R.K., Lad, K.B.: A review on side-view face recognition methods. Int. J. Innov. Res. Comput. Commun. Eng. **4**(3), 2984–2991 (2016)
6. David, H., Athira, T.A.: Improving the performance of SVM detection. In: Fourth International Conference on Advances in Computing and Communications, Kochi, 27—29 August 2014
7. Pallabi, P., Thuraisingham, B.: Face recognition using multiple classifiers. In: Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2006), May 2006
8. Athani, S., Tejeshwar, C.H.: Performance analysis of key frame extraction using SIFT and SURF algorithms. Int. J. Comput. Sci. Inf. Technol. **7**(4), 2136–2139 (2016)
9. Athani, S., Tejeshwar, C.H.: Content-based text retrieval using image processing techniques. Int. J. Comput. Sci. Inf. Secur. **14**(11), 556–561 (2016)
10. Ahonen, T., Hadid, A., Pietikäinen, M.: Face Description with local binary patterns: application to face recognition. IEEE Trans. Pattern Anal. Mach. Intell. **28**(12), 2037–2041 (2006)
11. Hall, L.O., Goldgof, D.B., Felatyev, S., Smarodzinava, V.: Horizon detection using machine learning techniques. In: Proceedings of the 5th International Conference on Machine Learning and Applications (ICMLA 2006) (2006)
12. Pujari, J.D., Yakkundimath, R., Byadgi, A.S.: Classication of fungal disease symptoms affected on cereals using colour texture features. Int. J. Signal Process. Image Process. Pattern Recogn. **6**(6), 321–330 (2013)
13. Burges, C.J.: A tutorial on support vector machines for pattern recognition. Knowl. Discov. Data Min. **2**, 121–167 (1998)

DEMST-KNN: A Novel Classification Framework to Solve Imbalanced Multi-class Problem

Ying Xia, Yini Peng, Xu Zhang(✉), and HaeYoung Bae

Research Center of Spatial Information System,
Chongqing University of Posts and Telecommunications, Chongqing, China
{xiaying,zhangx}@cqupt.edu.cn, pengyini740@163.com, hybae@inha.ac.kr

Abstract. Imbalanced multi-classification problem is an important research hotspot in machine learning. For multiple categories of data, an important method is divide-and-conquer, which covers the decomposition strategy and ensemble rule. However, with the increase of category, the number of poor binary-classifiers increases, which greatly reduces the classification accuracy. In this paper, a novel classification framework to solve current problem of multi-classification method for imbalanced data was proposed, which including a decomposition strategy based on maximum spanning tree, an ensemble rule based on node degree to find the optimal combination of binary-classifiers and a specific KNN algorithm which can dynamically adjust K neighborhood to increase probability of minority class. Finally we experimented on nine public data sets in different fields and compared our proposed classification framework with the most popular framework. The experimental results show that the proposed classification framework can greatly improve the classification accuracy of minority class.

Keywords: Classification framework · Multi-class · Imbalanced data · Decomposition strategy · Ensemble rule

1 Introduction

Imbalanced data has imbalanced distribution of various categories' instances. Data of many real-world applications such as bank credit risk assessment, network traffic anomaly detection, spam filtering, and cancer diagnosis are very imbalanced [1]. Therefore, classification learning of imbalanced data has become the focus of attention of the world's academic and industry. But at present, most of traditional classification algorithms are based on the premise of class distribution balance, such as SVM, KNN, and decision tree and so on [2]. Some of classification algorithm uses divide and conquer, but division of sample space will gradually lead to lack of minority class information and many inductive reasoning systems tend to classify samples for majority class in the presence of uncertainty [3]. These conditions greatly affected the classification accuracy.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_29

Some researches focus on constructing imbalanced multi-class classifier or extending original binary-classifier into imbalanced multi-class classifier to solve the problem. However, due to the variety and structure of real-world data, it is difficult to obtain a strong applicability of imbalanced multi-class classifier. Therefore, a divide-and-conquer approach is adopted to construct a multi-class decomposition framework [4]. The imbalanced multi-class problem is decomposed into a series of imbalanced binary classification problems, and then some ensemble rules are used to obtain the classification results. This approach covers key techniques such as decomposition strategies, sampling techniques, classification methods, ensemble rules, and so on. It has stronger portability and versatility, resulting in a series of related research and conclusions.

Solving imbalanced multi-class problem lies in taking into account the multiple categories and imbalance of data, and then improving classification accuracy, especially minority class.

There are many decomposition strategies to solve the multi-class problem of dataset, such as One versus All (OVA) method, One versus One (OVO) method and Decision Binary Tree (DB-Tree) method [5]. From the study [3] we can come to the conclusion that OVO scheme performs better other decomposition strategies in SVM, Decision Tree and Neural Network etc. But there is still a serious problem in OVO scheme that the number of binary classifier will grow exponentially along with increase of number of class. This will degrade the performance of the multi-class imbalance classification approach due to the increase of time consumption and bad classifiers' turnout.Besides, after process of decomposition, predicted values of each test sample are acquired from binary-classifiers, and the ensemble rule will be used to get the final classification result, including VOTE method, weighted voting and DDAG and so on [6]. These rules are applicable in some cases. In addition, the advantages and disadvantages of various ensemble rules were discussed in depth in the article [7], and it was found that there is no universal ensemble rule because of different data and classification method. Therefore different ensemble rules should be used for different situations.

The current solution to data imbalance is divided into two main areas. The first approach focuses on the data level, and sampling is the main technology. Another approach considers data imbalance in terms of algorithm level. These algorithms are optimized by considering misclassification cost, which leads to a good performance in the imbalanced data. However, these methods are the reconstruction for training set, and there is no emphasis for minority class in the classification algorithm. If we can pay more attention to the information of minority class in the classification process, it is more conducive to improve the classification accuracy of imbalanced data. Here we focus on K-Nearest Neighbor (KNN) algorithm to enhance classification accuracy of minority class according to dynamically adjusts division of test sample k-neighborhood.

Given that binary classifier is built between pairwise classes in OVO scheme, multi-class imbalanced data classification problem can be mapped into a complete graph when the class is denoted by node and the relationship between pairwise classes is expressed by edge. Due to the spanning tree contains all nodes of original graph and connects all nodes with the minimum number of edges, if edges of the graph can be assigned meaningful weight, it is easy to construct optimal spanning tree to find the minimum number of binary classifier.

In this paper, the main contributions of this paper can be summarized as follow:

(1) We propose a new imbalanced multi-classification framework to solve imbalanced multi-class problem, which includes a decomposition-ensemble approach based on maximum spanning tree to find the optimal combination of binary-classifiers and an improved KNN algorithm of adjusting minority class probability to weaken the impact of data distribution imbalance, so we named it DEMST-KNN.

(2) Experimental evaluation has been carried out over 9 public multi-class imbalanced datasets.

The rest of this paper is organized as follows. the processing of DEMST-KNN classification framework refer to Sect. 2; Sect. 3 completes experimental analysis from different perspective; the conclusion is summarized in Sect. 4.

2 Proposed Frameworks

2.1 An Imbalanced Multi-classification Framework

Multi-class and imbalance are important factors to solve the problem of imbalanced multi-classification. However, the most prevalent OVO scheme in multi-classification solutions has some limitations. Therefore, we can assume that OVO scheme can be used to mapping, and then the optimal combination of binary-classifiers can be obtained according to relationship between different categories. In addition, for imbalance for data, the adjustment of KNN neighborhood concept to enhance the probability of minority class, which can form our proposed new multi-classification framework - a decomposition-ensemble based on the maximum spanning tree and KNN algorithm with dynamic K-neighborhood (DEMST-KNN). Specific framework is as shown in Fig. 1.

2.2 Decomposition-Ensemble Based on the Maximum Spanning Tree (DEMST)

A preliminary step of graph-based method is to represent training data with an undirected weighted complete graph. For this purpose, given a training dataset P which have have n samples, m features and k classes, $P = \{p_1, p_2, ..., p_n\}$ where $p_i = \{f_{i1}, f_{i2}, ..., f_{im}\}$ and the class label is $C = \{C_1, C_2, ..., C_k\}$. A multi-class imbalance classification problem is mapped into its equivalent graph $G(V, E, W)$. The class labels is denoted by vertex set $V(G)$, edge set $E(G) = \{(C_i, C_j)|i, j = 1, 2, ..., k\}$ expresses inter-class relationships, and weight set $W(G) = \{\omega_{i,j}|i, j = 1, 2, ..., k\}$ represents inter-class distances.

Given a sample p, p is the core object when $N_\epsilon(p) \geq minPts$. $N_\epsilon(p)$ represents sample number in the ϵ neighborhood of p. $minPts$ is a positive integer, which expresses minimum number of objects threshold to be core object.

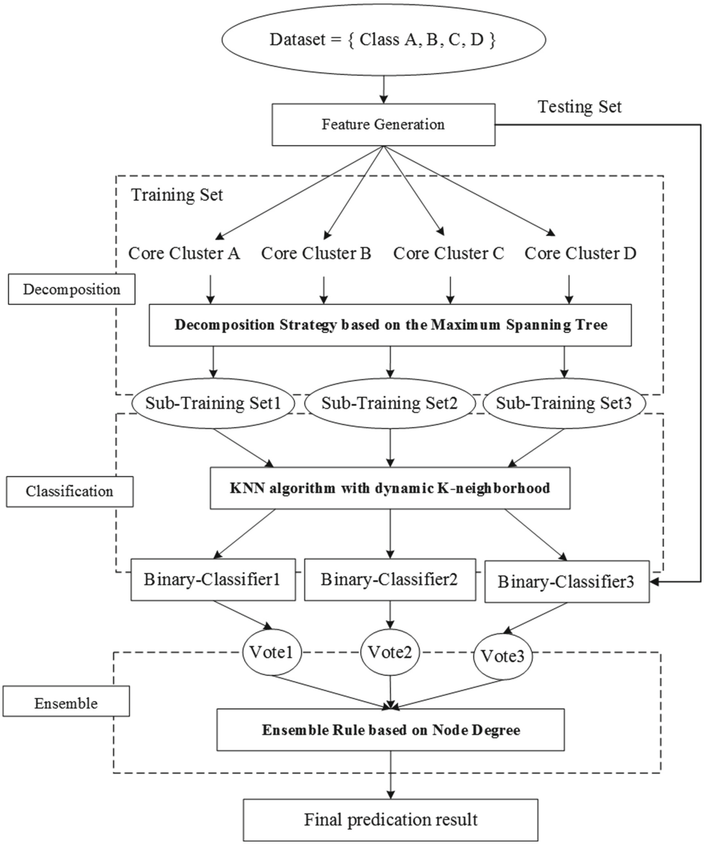


Fig. 1. Imbalanced multi-classification framework - DEMST-KNN.

Definition 1. Core cluster. The samples of Class i is denoted by $P_i = \{p_{i_1}, p_{i_2}, ..., p_{i_t}\} \subseteq P$, core cluster P_{i_r} meets several conditions: (1) $P_{i_r} \subseteq P_i$; (2) minimum number of nodes denotes as $minPts$, minimum radius of neighborhood is taken as ϵ, and $\forall q, w \in P_{i_r}$ in core cluster, if q is core object, from q to w is density-reachable about $minPts$ and ϵ.

If we calculate directly center position of class raw data, it is possible to go wrong due to outliers. So according to Definition 1, core cluster will be computed to determine the center position.

Definition 2. Inter-class distance. Assume that P'_{i_r} and P'_{j_r} represent center position of core cluster P_{i_r} and P_{j_r} respectively, inter-class distance is the Euclidean distance between P'_{i_r} and P'_{j_r}. If we set Dis function to express the Euclidean distance function, inter-class distance means weight between nodes, which is expressed as the following:

$$\omega_{i,j} = Dis(P'_{i_r}, P'_{j_r}) = \sqrt{\sum (x_i - y_i)^2} \tag{1}$$

Inter-class distance can be calculated according to formula (1), and multi-class imbalance classification problem is successfully changed to a weighted complete graph. Considering that the more inter-class distance between Class A and B, the easier trained classifier based on them can distinguish test sample. We put vertex of the graph into vertex of tree in turn to ensure that it has maximum distance between the two vertex sets. So the optimal classifiers are based on the edge set of maximum spanning tree. The above step can be summarized by the following pseudo-code of Algorithm 1.

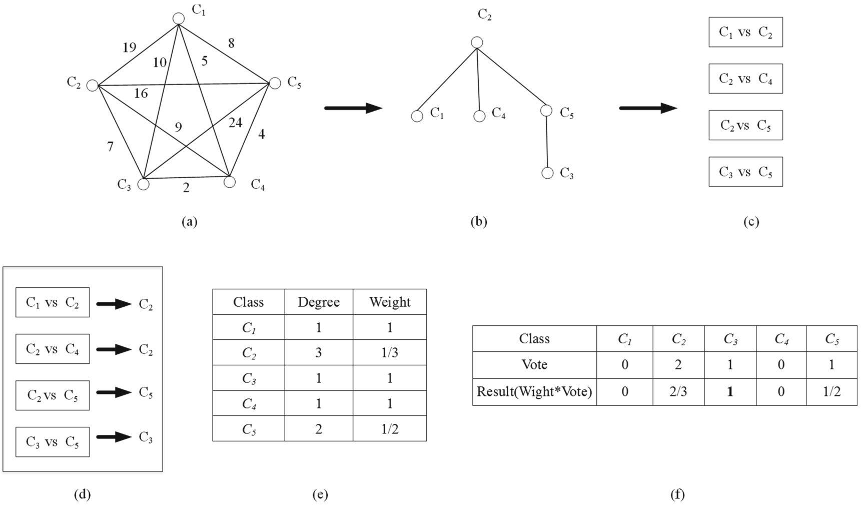
And next, it is obvious to understand that the degree of leaf node is 1 and other nodes are greater than 1. Therefore, a problem exists in that the class of leaf node only appears once in the construction of classifiers, but others are not. As a result, the weight of non-leaf class is greater than leaf class when we integrate all classification predication in some ensemble rule. And as such, the weight should be adjusted to keep every class having equal opportunity. The degree of class C_i is taken as $d(C_i)$, which expresses the number of edge associated with it. That is, $d(C_i)$ also denotes times of trained binary-classifier based on C_i. To guarantee the fair weight of classes, we define a weighted ensemble rule based on degree:

$$weight = 1/d(C_i) \tag{2}$$

The final prediction result depends on weight and vote of the class:

$$predict = 1/d(C_i) * vote \tag{3}$$

After ensuring equality of every class in this way, the classification predication will be improved significantly.



(a) (b) (c)

Class	Degree	Weight
C_1	1	1
C_2	3	1/3
C_3	1	1
C_4	1	1
C_5	2	1/2

Class	C_1	C_2	C_3	C_4	C_5
Vote	0	2	1	0	1
Result(Wight*Vote)	0	2/3	**1**	0	1/2

(d) (e) (f)

Fig. 2. The whole decomposition and ensemble process.

Algorithm 1. DEMST

Input: dataset P, class label $C = \{C_1, C_2, ..., C_k\}$
$P_i = \{p_{i_1}, p_{i_2}, ..., p_{i_t}\} \subseteq P$ and $1 \le i \le k$

Output: optimal sub-classifier: E'

```
procedure Finding the center cluster of each class
    for $i = 1$ to $k$ do
        $S = \oslash$;
        for each unvisited point $p$ in $P_i$ do        ▷ the label of points in $P_i$ is $C_i$
            $Q = \oslash$;
            make $p$ as visited;
            $Q = Q \cup p$;
            if isCorePoint($p$) then
                add directly density-reachable points of $p$ into $Q$;
                $S = S \cup \{Q\}$;
            end if
        end for
        for each pair $\{(\{S_j\}, \{S_t\}) | \{S_j\} \in S, \{S_t\} \in S$ and $j \ne t\}$ do
            if $S_j \cap S_t \ne \oslash$ then
                delete $\{S_j\}, \{S_t\}$ from $S$;
                $S_j = S_j \cup S_t$;
                $S = S \cup \{S_j\}$;
            end if
        end for
        $S_{i-core} = S_r \in S$; ▷ $S_r$ is element of $S$ which has most points and $S_{i-core}$ is the
            center cluster of class $i$
    end for
end procedure

procedure Calculate inter-class distance:$(Dif(C_i, C_j))$, and building a weighted
        complete graph:$(G(V, E, W))$
    for $i = 1$ to $k$ do
        calculate center position of $S_{i-core}$: $Position_i$
    end for
    for $i = 1$ to $k$ do
        for $j = 1$ to $k$ do
            $Dif(C_i, C_j) = Euclideandistance(Position_i, Position_j)$;
        end for
    end for
    $V(G) = \{C_1, C_2, ..., C_k\}$;
    $E(G) = \{(C_i, C_j) | i, j = 1, 2, ..., k\}$;
    $W(G) = \{Dif(C_i, C_j) | i, j = 1, 2, ..., k\}$;
end procedure

procedure Obtaining the maximum spanning tree maxtree:$(\mathrm{T}(V', E'))$ from
        graph:$(G(V, E, W))$
    $V' = \oslash$;
    $V' = V' \cup C_1$;
    while $V - V' \ne \oslash$ do
        choose $C_i$ from $V - V'$ which have a maximum weight between vertex $C_j$ in $T$
            and $C_i$;
        $V' = V' \cup \{C_i\}$;
        $E' = E' \cup \{(C_i, C_j)\}$;
    end while
end procedure
```

And to help understand DEMST, an instance of the above process is shown in Fig. 2. A dataset has 5 classes which are C_1, C_2, C_3, C_4, C_5. As Fig. 2(a) shows, a weighted complete graph was built when classes were as nodes and inter-class distance was weight of each edge. The maximum spanning tree in Fig. 2(b) was exported from graph, whose edges include (C_1, C_2), (C_2, C_4), (C_2, C_5), (C_3, C_5). Finally, 4 binary classifiers were trained on edge of the tree respectively as shown in Fig. 2(c), for example, training set of the first binary classifier is the sample of C_1, C_2. A test sample was respectively put into 4 binary classifiers of Fig. 2(c), and predications of 4 classifiers about the test sample are C_2, C_2, C_5, C_3 as Fig. 2(d) shown. The degree of every class can be obtained from Fig. 2(b), and then according to formula (2), we can calculate weight of every class which are 1, 1/3, 1, 1, 1/2 as Fig. 2(e) shown. Finally in Fig. 2(f), it is easy to get the updated prediction of the test sample on the basis of formula (3) and then the final result is determined Class C_3.

2.3 KNN Algorithm with Dynamic K-Neighborhood

In practical applications, the sample ratio between the minority and majority classes of an imbalanced dataset can be 1: 10, 1: 100, or even 1: 1000. In this case, the original KNN algorithm tends to over-compensate majority class and neglects minority class, so we consider define the dynamic k-neighborhood for nearest neighbor of test sample.

Definition 3. Dynamic K-neighborhood. When nearest neighbor of test sample is the minority class, we extend dynamic k-neighborhood of sample directly to the first boundary of majority class and minority class, as shown in Fig. 3(a). When nearest neighbor of test sample is the majority class, we extend dynamic k-neighborhood of sample to second boundary of majority class and minority class, as shown in Fig. 3(b). The training samples within the boundary are the dynamic k-nearest neighbors of test sample.

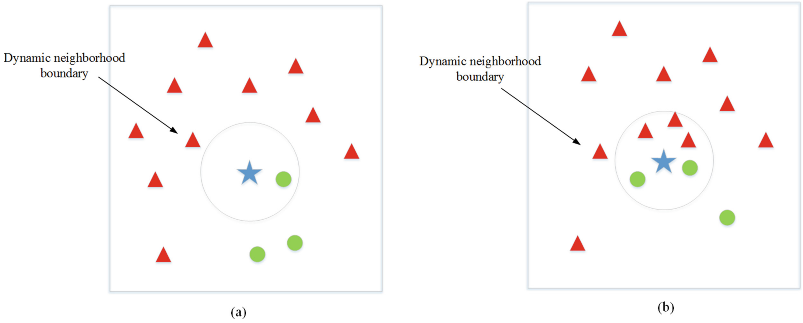


Fig. 3. The dynamic K-neighborhood boundary between majority class and minority class.

Algorithm 2. Dynamic K-neighborhood KNN
Input: Training set: $Tr = \{Tr_1, Tr_2, ..., Tr_n\}$
Testing set: $Te = \{Te_1, Te_2, ..., Te_n\}$
Output: probability of minority: P

```
procedure OBTAINING PROBABILITY OF MINORITY:(P)
    for each test sample $i$ in $Te$ do
        for each train sample $j$ in $Tr$ do
            $dis(j, 0) = Euclideandistance(i, j)$;
            $dis(j, 1) = j$;
        end for
        $sort(dis, 0)$;                                    ▷ Sorts the distance array
        $j = 0, r = 0, k = 0$;
        while $j < n$ do
            while $dis(j, 1) \in$ minority class do
                $r++; k++; j++$;
            end while
            if $j < n$ and $dis(j, 1) \notin$ minority class then
                break;
            end if
            while $dis(j, 1) \in$ majority class do
                $k++; j++$;
            end while
            while $dis(j, 1) \in$ minority class do
                $r++; k++; j++$;
            end while
            if $j < n$ and $dis(j, 1) \notin$ minority class then
                break;
            end if
        end while
        $\theta = (r/k)/(n_c/n)$;
        $P = (\theta * r + 1/n)/(\theta * r + (k - r) + 2/n)$;
    end for
end procedure
```

As shown in the Fig. 3, k is 1 in the Fig. 3(a), and the number of minority class is $r = 1$, and $k = 5$ in the Fig. 3(b), and the sample number of minority class is $r = 2$. We get a local neighborhood according to above definition, so the probability of current test sample for minority class can be converted to $P = (r + \frac{1}{n})/(r + (k - r) + \frac{2}{n})$.

Considering that the imbalance ratio in the local neighborhood and entire dataset are likely to vary greatly, we introduce an imbalance tilt factor θ to adjust the above-mentioned probability formula. θ is ratio of the local imbalance ratio and global imbalance ratio, that is $\theta = \frac{r}{k} : \frac{n_c}{n}$.

Adjusted final minority class probability for test sample:

$$P = \frac{\theta r + \frac{1}{n}}{\theta r + (k - r) + \frac{2}{n}} \tag{4}$$

The pseudo-code of dynamic K-neighborhood KNN is as shown in Algorithm 2.

3 Experiments and Evaluations

3.1 Experimental Implementation

Nine public datasets (http://archive.ics.uci.edu/ml/datasets.html) of different fields are selected to experiment. In order to verify the feasibility of the DEMST-KNN imbalanced multi-classification framework, we will compare it to the framework of the combination between the most widely used OVO decomposition strategy, VOTE ensemble rule and KNN classification algorithm. The reason why we choose it as comparing method is that OVO-VOTE has a significant boost for multi-class imbalanced data in the article [8]. A series of comparison experiments will adopt RUS and SMOTE as sampling technology to solve data imbalance and use KNN as classification algorithm. Experiment results will be analyzed in various perspectives. At last, classification accuracy is evaluated by k-fold cross validation. Considering time consumption of decomposition and sampling, 5-fold cross validation is adopted to get average accuracy (average 5-fold accuracy).

3.2 Experimental Evaluation

Figure 4 shows the average 5-fold overall classification accuracy for DEMST-KNN and OVO-VOTE based on RUS sampling with KNN classification algorithm (OVORUS-KNN) and SMOTE sampling with KNN classification algorithm (OVOSMOTE-KNN). Overall, DEMST-KNN has no significant advantage over other two to improve classification accuracy. On *sati.*, *car* dataset, we can see that classification accuracy of DEMST-KNN is even lower, but in general it still maintain the range of mean. Secondly, when *lymp.* and *glass* without enough samples adopts under-sampling technique, classification accuracy was decreased drastically. *sati.*, which has the largest number of samples, can still be classified with high accuracy regardless of whether it is sampled using under-sampling or SMOTE. The reason for this situation is likely to be under-sampling causes a serious loss of minority class information without enough samples, which makes it more difficult to distinguish.

And then we are taking classification accuracy analysis for the last two minority classes of each dataset as Fig. 5 shown. We can see that the three data sets, such as *lymp.*, *sati.* and *car*, OVORUS-KNN and OVOSMOTE-KNN cannot correctly identify minority classes, and DEMST-KNN achieves zero breakthrough because it can identify part of them. Overall, DEMST-KNN can significantly improve classification accuracy of minority class.

Most of imbalanced data real-life scenes pay more attention to information of minority classes, so here we propose the DEMST-KNN imbalanced multi-classification framework at the expense of majority class classification accuracy to improve the classification accuracy of minority class, thus maintaining the overall classification accuracy.

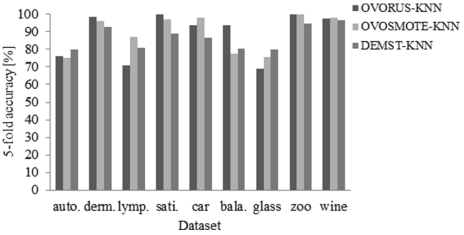


Fig. 4. Contrast bar graph: average 5-fold overall classification accuracy for DEMST-KNN, OVORUS-KNN and OVOSMOTE-KNN.

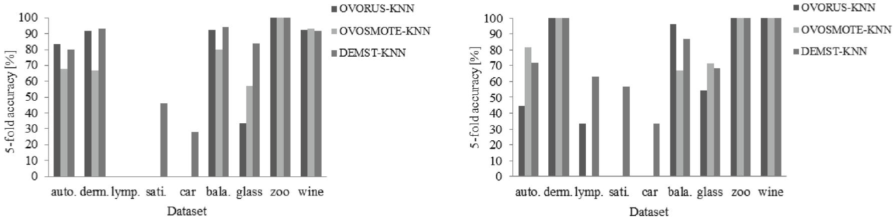


Fig. 5. Contrast bar graph: average 5-fold classification accuracy of minority class and second minority class for DEMST-KNN, OVORUS-KNN and OVOSMOTE-KNN.

4 Conclusions

In this paper, a new classification framework DEMST-KNN is proposed to solve imbalanced multi-classification problem. In this framework, we propose a decomposition strategy based on maximum spanning tree and an ensemble rule based on node degree to find the optimal combination of binary-classifiers, so as to avoid the problem that the number of poor binary-classifiers increases exponentially with the increase of category in the commonly used OVO strategy. Secondly, according to the imbalance of sample distribution among different classes, we dynamically adjust k neighborhood in KNN algorithm, thereby enlarging the probability of minority classes. In the end, we compare DEMST-KNN with the existing imbalanced multi-classification framework OVORUS-KNN and OVOSMOTE-KNN based on 9 public datasets. Experimental results show that DEMST-KNN maintains a fairly balanced state in the overall classification accuracy, and has no significant improvement compared with the other two methods. However, DEMST-KNN greatly improves the classification accuracy on minority classes. Therefore, DEMST-KNN is more effective to access to minority class information concerns for most realistic scenes such as network traffic anomaly detection, spam filtering, and cancer diagnosis.

Acknowledgments. This work is funded by Natural Science Foundation of China (41571401), Chongqing Natural Science Foundation (cstc2014kjrc-qnrc40002), Scientific and Technological Research Program of Chongqing Municipal Education Commission (KJ1500431), and WenFeng Creative Foundation of CQUPT (WF2014-05).

References

1. Sun, J., Lee, Y.C., Li, H., Huang, Q.H.: Combining B&B-based hybrid feature selection and the imbalance-oriented multiple-classifier ensemble for imbalanced credit risk assessment. Technol. Econ. Dev. Econ. **21**, 351 (2015)
2. Díez-Pastor, J.F., Rodríguez, J.J., García-Osorio, C., Kuncheva, L.I.: Random balance: ensembles of variable priors classifiers for imbalanced data. Knowl. Based Syst. **85**, 96 (2015)
3. Li, Y., Guo, H., Liu, X., Li, Y., Li, J.: Adapted ensemble classification algorithm based on multiple classifier system and feature selection for classifying multi-class imbalanced data. Knowl. Based Syst. **94**, 88 (2015)
4. Santhanam, V., Morariu, V.I., Harwood, D., Davis, L.S.: A non-parametric approach to extending generic binary classifiers for multi-classification. Patt. Recogn. **58**, 149 (2016)
5. Fernández, A., López, V., Galar, M., Jesus, M.J.D., Herrera, F.: Analysing the classification of imbalanced data-sets with multiple classes: binarization techniques and ad-hoc approaches. Knowl. Based Syst. **42**, 97 (2013)
6. Hüllermeier, E., Vanderlooy, S.: Combining predictions in pairwise classification: an optimal adaptive voting strategy and its relation to weighted voting. Patt. Recogn. **43**, 128 (2010)
7. Galar, M., Fernndez, A., Barrenechea, E.: An overview of ensemble methods for binary classifiers in multi-class problems: experimental study on one-vs-one and one-vs-all schemes. Patt. Recogn. **44**, 1761 (2011)
8. Zhang, Z., Krawczyk, B., Garca, S., Rosales-Prez, A., Herrera, F.: Empowering one-vs-one decomposition with ensemble learning for multi-class imbalanced data. Knowl. Based Syst. **106**, 251 (2016)

Combined Method for Integration of Heterogeneous Ontology Models for Big Data Processing and Analysis

Viktor Kureychik and Alexandra Semenova(✉)

Autonomous Federal State Institution of Higher Education,
Southern Federal University, Rostov, Russia
kur@tgn.sfedu.ru, alexaforum@rambler.ru

Abstract. In the given paper a combined method for integration of heterogeneous ontology for big data processing and analysis is proposed. This allows perform semantic search through heterogeneous information resources, represented by different ontologies. The fundamental difference of the proposed approach is that it allows obtaining optimal weights on the basis of which the optimal alignment of ontologies is carried out. Performed calculations validate the productivity of the proposed method.

Keywords: Big data · Ontology · Ontology alignment · Swarm intelligence · Multiobjective optimization

1 Introduction

Recent advances in scientific and technical areas, including computer and information technologies, have led to the fact that one of the main trends in the modern science development is a significant increase of experimental data volumes and the associated problems of storage and processing. It is clear that further successful development of research projects is possible only if the scientific community will learn to process and analyze extra-large amounts of data, and extract from them new knowledge. Formalization of unstructured data is one of the solutions for solving the problem of big data processing. So we may apply the formal ontology - a modern paradigm of computing resources that describe the knowledge of the world and subject areas.

Many Russian and foreign researchers investigated the problem of application of ontologies to the processing and analysis of big data. Nevertheless, the growth of unstructured information flows, the need to improve the quality of its analysis and processing in information systems requires the development of new methods for effective processing of big data from various domains. Shared ontologies accumulation is seen as a mechanism of unlimited accumulation of knowledge about the world. Currently the problem of comparing and matching ontologies at the level of alignment, i.e. finding semantic correspondences between the elements of two independently developed ontologies, is not solved yet. The problem of ontology alignment is to find such a structure and permissible parameters that provide the optimal values by one or more quality criteria.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_30

The purpose of this paper is to analyze the use of the ontological approach for big data processing and development of method for integration heterogeneous ontological models based on an evolutionary approach.

In this paper we propose a combined method for ontology alignment based on semantic similarity of concepts and multi-objective optimization of similarity weights. Modification of this method is the application of swarm intelligence algorithm for finding the weighting factors. The main advantages of the proposed approach are: finding the key concepts, eliminating of the subjectivity of their descriptions and dependence from the point of view of ontology developers. Generalized operation of concepts comparison along with the parsing and sorting algorithm will improve the quality of ontology alignment procedure. Therefore the interaction of heterogeneous information systems is provided. The fundamental difference of the proposed approach is that it allows obtaining optimal weights on the basis of which the optimal alignment of ontologies is carried out. Performed calculations validate the productivity of the proposed method.

2 The Problem of Unstructured Information Integration

The concept «BigData» refers to the data sets of extremely large volume and complexity that standard tools are not able to carry out their capture, storage, management and processing within a reasonable time for practice. Big data is characterized by parameters such as [1]:

- volume: big data doesn't sample; it just observes and tracks what happens;
- velocity: big data is often available in real-time;
- variety: big data draws from text, images, audio, video; plus it completes missing pieces through data fusion;
- validity: the property of being genuine, a true reflection of attitudes, behavior, or characteristics.

Unstructured information (NI) is information that either does not have a predetermined data structure or not organized in the given order [1]. Ontology is a formal explicit description of classes (concepts) in the domain, the properties and attributes of each concept (slots), and the restrictions imposed on slots (facets) [3]. Domain structure (ontology) development is the first step to bring the NI to a structured form. Each individual domain is only a subset of unstructured data set, so for the best possible coverage of the data and, as a consequence, a more complete analysis is necessary to allocate the maximum possible number of different domains to be analyzed [4].

Nowadays the heterogeneous information systems accumulate a considerable amount of knowledge. While integration of these systems a problem of classifying and structural representation of knowledge from different domains appears [5]. Different contexts of ontologies created by different communities are reflected in the different approach to the concepts of the specification that has become one of the causes of heterogeneity. As a result, the semantics of the concepts in the contexts described by different ontologies may be similar in different approaches of description of their structure: the structure, constraints and level of detail.

Linguistic approach for ontology integration involves the creation of a formal ontology of the upper level, the interaction of which with other ontologies is implemented on the basis of linguistic relations. Linguistic relations of such ontological models such as *synonym_of* (synonym), *hyponym_of* (gipernim), *overlap_of* (overlap) and other linguistic relationships allow formally implement the mapping of terms [6]. The disadvantage of this approach is that the linguistic relationships are not always adequately reflect the semantics because of the ambiguity of the language variables.

Ontology integration approach based on shared vocabulary allows to build an integrated model of the different domains of knowledge due to the fact that almost any notion of a vocabulary can be associated with any other term. However, in this case the integration of ontologies typically performed with some ontology developer limits and tips. This approach is implemented by viewing the two ontologies, finding in them synonyms, as well as by conflicts resolution and the creation of a third ontology.

Integration of heterogeneous ontologies may be performed based on alignment of instances: the semantic relationships between two classes of heterogeneous ontologies are merged on the basis of the intersection of sets of instances. Typically, ontology classes described multiple instances that allow better define the semantics of a class. Therefore, the association of ontologies based on instances is more effective [7].

The main disadvantage of the majority of unstructured data fusion methods is the need to engage an expert to confirm the correctness of the detection of the similarities and differences of semantic concepts. Thus, ontological approach provides a new level of information integration. For semantically correct interconnection of heterogeneous information systems it is necessary to compare ontology and to find out their differences and similarities. This problem is solved by semantic similarity techniques of concepts of ontonologies.

3 Ontology Integration Based on Semantic Similarity

An approach for integrating unstructured data based on a comparison of the results of concepts, their attributes and relationships between concepts on the level of ontology alignment is suggested [8]. Each concept of the domain ontology is defined as a unit of knowledge and identified by a name and a type. We define concept as [8]

$$C_i = (N_i, T_i), \tag{1}$$

where

- N_i – a unique name (identifier) of i-th concept;
- T_i – a type of i-th concept.

Let's $C = \{C_i | i = 1, 2, \ldots, n\}$ be a set of concepts and $\mathrm{R} = \{\mathrm{R}_1, \mathrm{R}_2, \mathrm{R}_3\}$ a set of relations between concepts. At that,

- $R1$– relation of inheritance (relation of «class-subclass»), $R1(C1,C2)$, where $C1$ – is a superclass of $C2$;
- $R2$ – relation of aggregation (relation of «whole-part»), $R2(C1,A')$: attributes of concept $C1$ are included in a set of attributes of all concepts A';

- *R*3 – relation of association (semantic relations), having transitive relation.

Let's consider the following expression of formal ontology [9]:

$$\mathrm{OHT} = (C, P, R, A), \tag{2}$$

where

- C – denotes concept (or classes) set for a specific domain;
- P – set of concepts attributes. Property is a component of the relation p(c,v,f), where c ∈ C – ontology concept, v – property value, associated with c and f defines restrictions for facets in v. One of restriction is a type, capacity, and range.
- R = {r | r ⊆ C × C × R t} – set of binary relations between concepts in C. There is the following variety of relation types: 1:1, 1:many, many:many. The basic set of relations are: synonymOFF, kindOFF, partOFF, instanceOFF, propertyOFF.
- A – axioms' set. Axiom is a rule that specify cause-and-effect relationship.

The problem of heterogeneous ontology combination is formulated as follows: given two regular ontologies create a third regular ontology, which is the concept of the input ontologies, as well as additional restrictions and relationships, if they are required. Building an ontology mapping O_1 on O_2 ontology is to find for each concept of ontology O_1 similar to it concept of ontology O_2.

Different ontologies may have overlapping sets of attributes, relations and concepts. The resulting ontology, maintaining the specifications in such a way as to include all the possible relations between concepts and did not contain equivalent (duplicate) concepts is developed on the basis of multiple source ontologies. So mappings on the same concepts of ontologies match. The resulting ontology defines the concepts of compliance and interpretation of the rules that can successfully establish their interaction. The purpose of the integration of unstructured data is to maintain compliance of the set of ontologies to the defined set of semantic relations. Semantic relations defined on the ontology O is taken as z-predicate set on O'. If there is a semantic relation z in ontology O, we write z (O).

Initially heterogeneous ontologies are not associated between each other. Therefore we need to find semantically similar elements of ontologies. For the numerical evaluation of semantic similarity of ontology concepts an approach based on the results of studies of A.F. Tuzovskiy was chosen [10]. In the proposed method similarity measure consists of three components: attributive, taxonomic and relational measures. This method has been adapted for the calculation of the semantic similarity of two heterogeneous ontologies. Modification of this method is an application of particle swarm algorithm for finding the weights. The definition of lexical component is calculated as the ratio of the intersection of sets of words (synonyms) in terms of their association.

Let's $S^T(c_i, c_j), S^R(c_i, c_j), S^A(c_i, c_j)$ be a semantic measure of two concepts based on their position, attribute concept attribute value. Weights t, allow to control commutation process of semantic similarity of two concepts.

To estimate lexical similarity of two concepts $S^T(c_i, c_j)$ sets of concept terms are $\mathrm{PL}_p(c_i)$ and $\mathrm{PL}_p(c_j)$ compared, common and different components are found [8]:

$$S^T(c_i, c_j) = \begin{cases} 1, \text{если } c_i = c_j \\ \frac{|PL_p(c_i) \cap PL_p(c_j)|}{|PL_p(c_i) \cup PL_p(c_j)|}, \quad \text{если } c_i \neq c_j \end{cases} \tag{3}$$

where $PL_p(c_i) = \{L_i \in L | P_c(c_i) = L_i\}$ – is a set of lexical terms of a concept c_i.

To estimate relation similarity it is assumed that if two concepts have similar relation with the third concept they are more similar than two concepts having different relations. Let's assume that [8]

$$\mathrm{C}_r(c_i) = \{c_j \in C | R_1(c_i, c_j) \vee R_2(c_i, c_j) \vee R_3(c_i, c_j) \vee c_j = c_i\} \tag{4}$$

− is a set containing concepts with relations R_1, R_2, R_3.

Define association relation of concepts such as [8]

$$R_A(c_j) = \{c_i : c_i \in C_r(c_j)\}. \tag{5}$$

Calculate the sum lexical similarity values for the set of concepts (c_j) and $R(c_i)$.

$$S_{RA}(R_A(c_i), R_A(c_j)) = \sum\nolimits_{c_i \in R_A(c_i), c_j \in R_A(c_j)} S^T(c_i, c_j) \tag{6}$$

Relation similarity measure $S^R(c_i, c_j)$ allows to evaluate similarity of two concepts based on concept similarity from a set (c_i) [8].

$$S^R(c_i, c_j) = \begin{cases} 1, \quad \text{если } c_i = c_j \\ \frac{S_{RA}(R_A(c_i), R_A(c_j))}{|R_A(c_j) \cup R_A(c_i)|} \quad if\ c_i \neq c_j \end{cases}$$

Compare attributes of two concepts. A set of attributes pertaining to a concept:

$$A^{Ci} = \{A_k^{Ci}, k \in [1\ldots n_1]\}, \tag{7}$$

where n_1 – number of attributes in a concept c_i.

$$A^{Cj} = \{A_k^{Cj}, k \in [1\ldots n_2]\}, \tag{8}$$

where n_2 − number of attributes in a concept c_j.

Attributive similarity measure $S^A(c_i, c_j)$ of concepts c_i and c_j is calculated by matching of common attributes: $A^{C_i} \cap A^{C_j}$. Attributive similarity measure $S^A(c_i, c_j)$ satisfy axioms of independence and resolvability, and is defined by the expression

$$S^A(c_i, c_j) = \frac{|A^{C_i} \cap A^{C_j}|}{|A^{C_i} \cup A^{C_j}|}, \tag{9}$$

where A^{C_i} – is a set of attributes of a concept c_i;

A^{C_j} − is a set of attributes of a concept c_j.

Similarity measure S $\left(c_i, c_j\right)$ of concept c_i of ontology O and concept c_j of ontology O' is defined

$$S\left(c_i, c_j\right) = t \cdot S^T\left(c_i, c_j\right) + r \cdot S^r\left(c_i, c_j\right) + \alpha \cdot S^A\left(c_i, c_j\right) \tag{10}$$

where t, r, a − are the coefficients, defining importance of similarity measures $S^T\left(c_i, c_j\right), S^R\left(c_i, c_j\right), S^A\left(c_i, c_j\right)$, respectively,

$$t, r, \alpha \in [0; 1], t + r + \alpha = 1, S\left(c_i, c_j\right) \in [0; 1]. \tag{11}$$

$$\begin{cases} S\left(c_i, c_j\right) = 1, & \textit{if concepts are equivalent}, \\ S\left(c_i, c_j\right) = 0, & \textit{if concepts are different}. \end{cases}$$

Heterogeneous ontology integration problems belong to a class of NP-hard optimization problems, and can be solved by evolutionary algorithms.

4 Multi-objective Optimization of Similarity Weights Calculation

Consider the modified swarm intelligence method for ontology alignment, using multi-objective optimization approach [11]. Algorithm for optimization of similarity weights calculation by particle swarm intelligence is depicted on Fig. 1 [12].

In this work we propose to apply PSO calculation based on multi-objective optimization. Multi-objective optimization or parallel programming is the process of simultaneous optimization of two or more conflicting objective functions in a given domain. Multi-criteria optimization task is formulated as follows [13]:

$$\min_{\vec{x}}\{f_1(\vec{x}), f_2(\vec{x}), \ldots, f_k(\vec{x})\}, \quad \vec{x} \in S \tag{12}$$

where $f_i : R^n \to R$ − is a $k(k \geq 2)$ of objective functions. Solution vectors $\vec{x} = (x_1, x_2, \ldots, x_n)^T$ belong to a non-empty domain set S.

Multi-objective optimization task is to find a vector of target variables satisfying cash constraints and optimizing the function of the vector whose elements correspond to the objective function. These functions form a mathematical description of the satisfactory test [14].

Consider the set of data, wherein the data lines are different similarity coefficients and columns - the relations between two different ontologies. For subsequent combining these similarity coefficients in one metric optimum weights were obtained. The proposed approach is possible to find the set of weights that meet the criteria of similarity which allows obtaining an optimal alignment. In the process of evaluation of swarm intelligence generalized function was calculated $\mathrm{f}_{\mathrm{integ}}$:

$$f_{\mathrm{integ}}(O1_i, O2_i) = \sum_{k=1}^{7} w_k \times F_k\left(salign_{ij}\right), \tag{13}$$

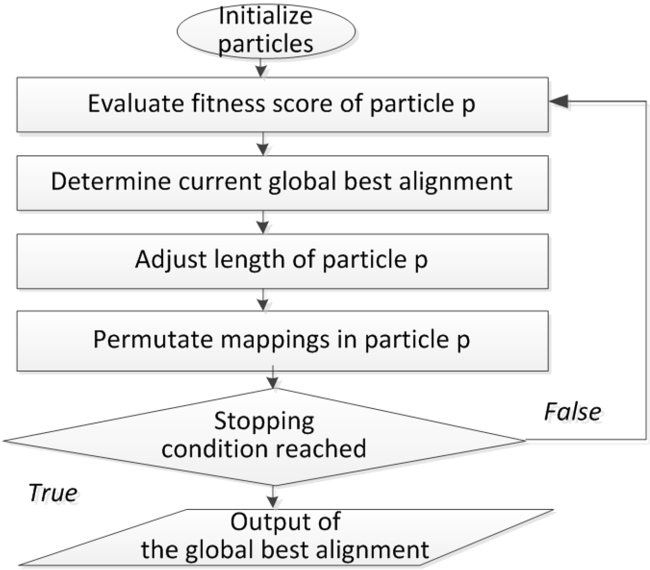

Initialize particles
Evaluate fitness score of particle p
Determine current global best alignment
Adjust length of particle p
Permutate mappings in particle p
Stopping condition reached
False
True
Output of the global best alignment

Fig. 1. Algorithm for optimization of similarity weights by particle swarm intelligence

where $\sum_{k=1}^{7} w_k = 1$.

If $f_{integ}(O1_i, O2_i)$ exceeds a threshold then $salign_{ij}$ is a valid alignment. In such a way all valid alignments are defined. Subsequently, using these valid alignments and reference alignments objective functions are calculated.

The method consists of the following stages.

(A) Initialization. The population is called a swarm, and it is composed of m number of appropriate solutions or particles. Each particle has n positions or cells comprising n weighting coefficients corresponding to n different similarity measures. Initially, for each cell of particles the value from 0 to 1 is selected randomly. Once selected primary swarms, calculated the corresponding values of fitness. The initial velocity of the particles of each cell is zero. The inputs to the proposed method are a swarm of 50 and weighting factors c_1 and c_2. The threshold value is chosen to be 0.5. The algorithm performed within 30 iterations.

(B) Objective function. The proposed approach works with multiple objective functions: the accuracy and recall of the search. Accuracy is the criterion of correct alignment found in the resulting alignment. Recall is the criterion of finding the right alignment found from a given reference alignment. The criterion of «accuracy» is calculated by the following formula:

$$P = \frac{|A| - |A \cap R|}{|A|} \tag{14}$$

The criterion of «recall» (quantitative parameter of the results of information retrieval, which is determined by dividing the amount granted as a result of the search of relevant concepts to the total number of relevant concepts present in the ontological model) is calculated by the following formula

$$R = \frac{|A| - |A \cap R|}{|A|} \tag{15}$$

As proposed multi-objective Particle Swarm Optimization is implemented as minimization problem so first objective is computed as (1-precision) and second objective is computed as (1-recall).

(C) Next Generation Swarm is Produced by Evaluating the Position and Velocity. Each cell or position represents the weight (normalized value of the cell) with respect to the similarity measure. The cells inside the particles contain values from 0 to 1, and the speed of each gene is given zero values. Using the information obtained in the previous step, the position and velocity of each particle of each cluster are updated. Each particle keeps track of the best position it has reached, which is also called pbest. In terms of multi-criteria approach, the position is selected for pbest, whose adaptation of the particle dominates the other devices. And the best position among all particles called global best or gbest. When the particle moves to a new position at a rate that its position and velocity changes in accordance with Eqs. (16) and (17) [13]:

$$v_{ij}(t+1) = w \times v_{ij}(t) + c_1 \cdot r_1 \cdot \left(pbest_{ij}(t) x_{ij}(t)\right) + c_2 \cdot r_2 \cdot \left(gbest_{ij}(t) - x_{ij}(t)\right) \tag{16}$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \tag{17}$$

where t is a time stamp, j-th cluster of i-th particle. Velocity $v_{ij}(t+1)$ is calculated by usage of previous velocity $v_{ij}(t)$, pbest and gbest. Then a new position $x_{ij}(t+1)$ is obtained by adding new velocity with current position $x_{ij}(t)$. c_1 and c_2 are set to 2, r_1 and r_2 are random values from the range from 0 to 1.

After applying non-dominated sorting and crowding distance sorting to the archive, a Local Search is conducted for obtaining the better approximation of weights regarding optimal alignment. In the Local-Search algorithm, the best particle replaces the worst particle of the new generation.

5 Experimental Research

Experimental researches performed with different number of ontology entities have shown that the algorithm has polynomial time complexity $O(n^2)$. Diagram of time complexity of the algorithm is shown on Fig. 2.

We've compared the suggested approach with single objective optimization by accuracy and recall (Table 1) [7].

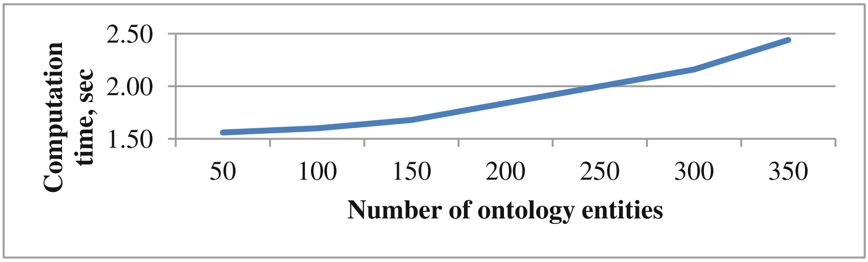


Fig. 2. Diagram of time complexity of the algorithm

Table 1. Results of experiments

Method	Accuracy	Recall	F-measure
Multi-objective optimization	0,8957	1,00	0,9873
Optimization by one objective function (accuracy)	0,7856	0,946	0,8583
Optimization by one objective function (recall)	0,3666	1,00	0,0103

Efficiency of the suggested approach for the criterion accuracy is 0,81428 (high). Again with respect to the f-measure the table shows that our method outperforms other single objective versions. Therefore, the proposed method is effective.

6 Conclusion

The ontological approach for big data processing is considered. The approach for integrating unstructured data is based on comparison of the results of concepts, their attributes and relationships between concepts on the level of ontology alignment. Each concept of the domain ontology is defined as a unit of knowledge and identified by a name and a type. The purpose of the integration of unstructured data is to maintain compliance of the set of ontologies to the defined set of semantic relations. Heterogeneous ontology integration problems belong to a class of NP-hard optimization problems, and can be solved by evolutionary algorithms. In this work we propose to apply PSO calculation based on multi-objective optimization. Experimental researches performed with different number of ontology entities have shown that the algorithm has polynomial time complexity.

The main advantages of the proposed approach are: finding the key concepts, eliminating of the subjectivity of their descriptions and dependence from the point of view of ontology developers. Generalized operation of concepts comparison along with the parsing and sorting algorithm will improve the quality of ontology alignment procedure. Therefore the interaction of heterogeneous information systems is provided. The fundamental difference of the proposed approach is that it allows obtaining optimal weights on the basis of which the optimal alignment of ontologies is carried out. Performed calculations validate the productivity of the proposed method.

References

1. Analysis of unstructured data: applications of text analytics and sentiment mining. Elektronnyj resurs, data obrashcheniya aprel (2016). https://support.sas.com/resources/papers/proceedings14/1288-2014.pdf
2. Lapshin, V.: Ontologii v komp'yuternyh sistemah. Nauchnyj mir, Moskva (2010)
3. Gruber, T.R.: The role of common ontology in achieving sharable, reusable knowledge bases. In: Allen, J.A., Fikes, R., Sandewell, E. (eds.) Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning, pp. 601–602. Morgan Kaufmann, Cambridge (1991)
4. Ontologicheskie metody i sredstva obrabotki predmetnyh znanij: monografiya. Palagin, A. V., Kryvyj, S.L., Petrenko, N.G. - Lugansk: izd-vo VNU im. V. Dalya, 324 s (2012)
5. Kopajgorodskij, A.N. Primenenie ontologij v semanticheskih informaci-onnyh sistemah. Ontologiya proektirovaniya № 4(14), str.90–98 (2014)
6. Semenova A.V., Kurey, F. [chik V.M. Obzor metodov analiza i obrabotki lingvi-sticheskoj ehkspertnoj informacii. Informatika, vychislitel'naya tekhnika i inzhenernoe obrazovanie, № 1 (12). S. 25–77 (2015)
7. Euzenat, J., Shvaiko, P.: Ontology Matching. Springer, Heidelberg (2007)
8. Bubareva, O.A.: Matematicheskaya model' processa integracii informacionnyh sistem na osnove ontologij. Bubareva, O.A., Popov, F.A. Sovremennye problemy nauki i obrazovaniya, № 2 (2012). http://www.science-education.ru/102-6030. Data obrashcheniya 19 April 2016
9. Semenova, A.V., Kureychik, V.M.: Domain ontology development for linguistic purposes. In: 9th International Conference on Application of Information and Communication Technologies (AICT) (2015)
10. Tuzovskij, A.F.: Metod obedineniya ontologij predmetnyh oblastej znanij. Izvestiya Tomskogo politekhnicheskogo universiteta, T 309, № 7, C. 138–141 (2006)
11. Bock, J.: Ontology alignment using biologically-inspired optimisation algorithms. Dissertation, Karlsruher Institut fur Technologie (KIT) Fakultut fur Wirtschaftswissenschaften (2012)
12. Semenova, A.V., Kureychik, V.M.: Application of swarm intelligence for domain ontology alignment. In: Abraham, A., Kovalev, S., Tarassov, V., Snášel, V. (eds.) Proceedings of the First International Scientific Conference "Intelligent Information Technologies for Industry" (IITI 2016). Advances in Intelligent Systems and Computing, vol. 450, pp. 261–270. Springer, Cham (2016)
13. Semenova, A.V., Kureychik, V.M.: Multi-objective particle swarm optimization for ontology alignment. In: 10th International Conference on Application of Information and Communication Technologies, pp. 141–147 (2016)
14. Gladkov, L.A., Kureychik, V.V, Kureychik V.M.: Bioinspirirovannye metody v optimizacii: monografiya. - M: Fizmatlit, S. 384 (2009)

Enhancing Extraction Method for Aggregating Strength Relation Between Social Actors

Mahyuddin K.M. Nasution(✉) and Opim Salim Sitompul

Information Technology Department, Fakultas Ilmu Komputer Dan Teknologi
Informasi (Fasilkom-TI), and Information System Centre,
Universitas Sumatera Utara, 1500 USU, Medan, Sumatera Utara, Indonesia
mahyuddin@usu.ac.id

Abstract. There are differences in the resultant of extracting the relations between social actors based on two streams of approaches in principle. However, one of the methods like the superficial methods can upgraded to make the information extraction by using the principles of the other methods, and this needs proof systematically. This paper serves to reveal some formulations have the function for resolving this issue. Based on the results of experiments conducted the expanded method is the adequate.

Keywords: Search engine · Search term · Query · Social actor · Singleton · Doubleton

1 Introduction

Extracting social network from Web has carried out with a variety of approaches ranging from simple to complex [1]. Unsupervised method or superficial method generally more concise and low cost, but only generates the strength relations between social actors from heterogeneous and unstructured sources such as the Web [2]. Instead, supervised methods are generally more complicated and high cost and it produces labels of relationship between social actors, but it came from sources, homogeneous and semi-structured like corpuses [3,4]. However, to generate social networks that enable to express semantically meaning is not easy [5]. This requires a method to represent their privilege of both methods: An approach is not only produces a relationship but re-interpret the relationship based on the aggregation principle. This paper aimed to enhance the superficial method for extracting social network from Web.

2 Problem Definition

The initial concept semantically of the extraction of social network from Web is to explore a series of names through co-occurrence using search engine [6,7]. Then, the extraction of social network made possible by involving the occurrence. Formally, the following we stated extracting social networks [8,9].

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_31

Definition 1. *Let* $A = \{a_i | i = 1, \ldots, n\}$ *is a set of social actors. The social network extraction* (SNE) *is* $\langle A, V, R, E, \gamma_1, \gamma_2\rangle$ *with the conditions as follows*

A1 $\gamma_1 : A \overset{1:1}{\rightarrow} V$, *and*
A2 $\gamma_2 : A \times A \rightarrow E$.

where $V \neq \emptyset$, $V = \{v_i | i = 1, \ldots, n\}$ *is a set of vertices in* G *and* $E = \{e_j | j = 1, \ldots, m\}$ *is a set of edges in* G, *or* $G = \langle V, E\rangle$ *as graph, and* $e_j = r_s$ *in* R, R *is a set of relations.*

Occurrence and co-occurrence individually are a query (q) representing a social actor and a query representing a pair of social actors. On the occurrence, q contains a name of social actor, for example $q =$ "Mahyuddin K. M. Nasution". While on the co-occurrence, q contains two names of social actors, for example $q =$ "Mahyuddin K. M. Nasution", "Shahrul Azman Noah" [2]. Therefore, names of social actors are the search terms, and we define it formally as follows

Definition 2. *A search term* t_k *consists of words or phrase, i.e.* $t_k = \{w_k | k = 1, \ldots, o\}$.

We use the well query to pry information from the Web by submitting it to search engine. A search engine works on a collection of documents or web pages, or more precisely as follows [10].

Definition 3. Ω *is a set of web pages indexed search engine, if there are a table relation of* (t_i, ω_j) *such that* $\Omega = \{(t, \omega)_{ij}\}$, *where* t_i *is search terms and* ω_j *is a web pages,* $i = 1, \ldots, I$, $j = 1, \ldots, J$. *The cardinality of* Ω *is denoted by* $|\Omega|$.

Definition 4. *Let* t_x *as search term,* $\Omega_x \subseteq \Omega$ *is a singleton space of event if*

$$\Omega_x(t_x) = \begin{cases} 1 & if\ t_x\ is\, true\ at\ \omega \in \Omega, \\ 0 & otherwise. \end{cases}$$

Briefly we denote it as $\omega \Rightarrow t_x$ *is true.*

Proposition 1. *If* $t_x \in \mathcal{S}$, $\mathcal{S}$ *is a set of singleton search terms of search engine, then there are vector space* $\Omega_x \subseteq \Omega$ *for generating hit count* $|\Omega_x|$.

Proof. For $t_x \in \mathcal{S}$, it means that $t_x \in \omega$ and $\omega \in \Omega$, or $\mathcal{S} \in \Omega$, every web page that is indexed by search engine contains at least one occurrence of t_x, where based on Definition 3 $\Omega_x = \{(t_x, \omega_x)_{ij}\}$. Thus for all web pages be in force $\{\Omega_x(t_x) = 1\}$ so that $\Omega_x \subseteq \Omega$ be the singleton search engine event of web pages that contain an occurrence of $t_x \in \omega_x$, and based on Definition 4, we have $\Omega_x = \{(\omega_x \Rightarrow t_x)\}$ as vector space. We call $\Omega_x \subseteq \Omega$ as *singleton*. The cardinality of Ω_x be $|\Omega_x|$ is the hit count of singleton as follows

$$|\Omega_x| = \sum_{\Omega} (\Omega_x(t_x) = 1). \tag{1}$$

It proved.

As information of any social actor, the singleton is the basic of search engine property that statistically related to the social actor. In this case, the singleton be the necessary condition for gaining the information of social actor from Web although it contains connatural trait (bias and ambiguity), and naturally it becomes the social dynamic of human beings [2]. Hit count is main information for a social actor based on Web, and validation of this information can obtained by crawling one after one the snippets list returned by the search engine [11].

Definition 5. *Let t_x is a search term. A snippet is a set of words, i.e. $s_x = \{w_i | i = 1, \ldots, \pm 50\}$. A list of snippets is $\mathcal{L}_x = \{s_j | j = 1, \ldots, J\}_x$, $m \leq |\Omega_x|$, then size of snippet $|s_x| = \pm 50$ words and $\mathcal{L}_x$ is a matrix M_{jk} that consists of j rows and k columns where each column refers to one token of word $w \in s_j$.*

Lemma 1. *If w is a token in $\mathcal{L}$, then w statistically has the character.*

Proof. It is known that $w \in s_x$, $s_x \in \mathcal{L}$, we have w as one token in $\mathcal{L}$. Based on Proposition 1 we have the hit count $|\Omega_x|$. Let $w \in M_k$ and if $m_{jk} \in M_{jk}$ we obtain

$$m_{jk} = \begin{cases} 1 & \text{if } w \in s_j \\ 0 & \text{otherwise,} \end{cases}$$

Therefore, at column w we obtain $|w| = \sum_{j=1}^{J} m_{jk}$ as the weight of w, and proved that w has the character, i.e. the relative probability of w

$$p(w) = \frac{|w|}{|\Omega_x|} \in [0, 1], \tag{2}$$

where $|w| \leq |\Omega_x|$ and $|\Omega_x| \neq 0$.

Definition 6. *Suppose t_x and t_y is two search terms, $\Omega_x \cap \Omega_y \subseteq \Omega$ is a doubleton space of event if*

$$\Omega_x(t_x \wedge t_y) = \Omega_y(t_x \wedge t_y) = \begin{cases} 1 & \textit{if } t_x \textit{ and } t_y \textit{ are true at } \omega \in \Omega, \\ 0 & \textit{otherwise} \end{cases}$$

where $\Omega_x \subseteq \Omega$ and $\Omega_y \subseteq \Omega$. Briefly we denote it as $\omega \Rightarrow t_x \wedge t_y$ is true.

Proposition 2. *If $t_x, t_y \in \mathcal{D}$, $\mathcal{D}$ is a set of doubleton search term of search engine, then there are vector space $\Omega_x \cap \Omega_y \subseteq \Omega$ for generating hit count $|\Omega_x \cap \Omega_y|$.*

Proof. Similar to Proposition 1, and based on Definitions 3 and 6, we have the hit count of doubleton as follows

$$|\Omega_x \cap \Omega_y| = \sum_{\Omega} (\Omega_x(t_x \wedge t_y) \cap \Omega_y(t_x \wedge t_y)) = 1). \tag{3}$$

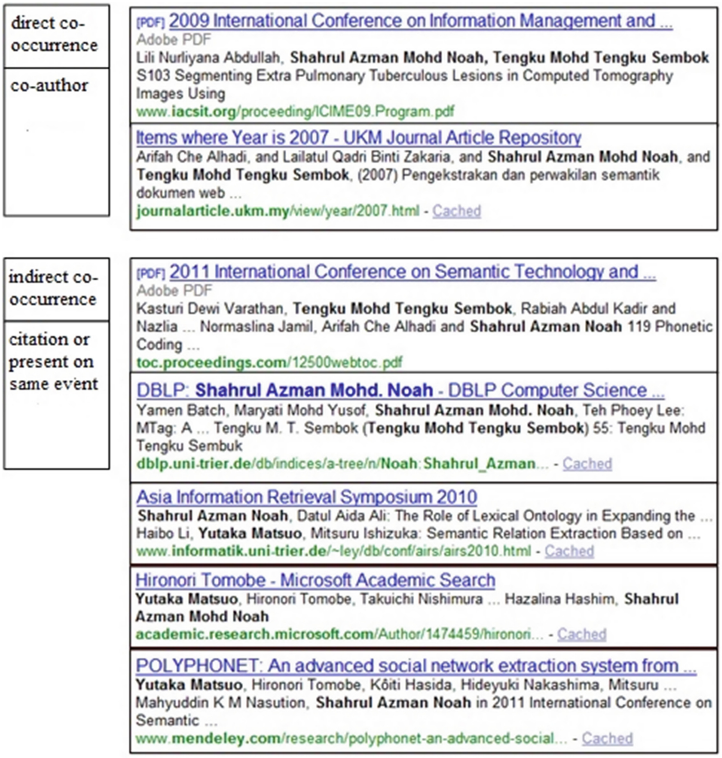
It proved.

Lemma 2. *If w is a token in $\mathcal{L}_D$ as list of snippets based on doubleton, then w statistically has the character in the doubleton.*

Proof. Similar to Lemma 1, and based on Definitions 5 and 6, we have the character of w in the doubleton as follows

$$p_{\mathcal{D}}(w) = \frac{|w|}{|\Omega_x \cap \Omega_y|} \in [0,1], \tag{4}$$

where $|w| \leq |\Omega_x \cap \Omega_y|$ and $|\Omega_x \cap \Omega_y| \neq 0$.



direct co-occurrence

co-author

[PDF] 2009 International Conference on Information Management and ...
Adobe PDF
Lili Nurliyana Abdullah, **Shahrul Azman Mohd Noah, Tengku Mohd Tengku Sembok** S103 Segmenting Extra Pulmonary Tuberculous Lesions in Computed Tomography Images Using
www.**iacsit.org**/proceeding/ICIME09.Program.pdf

Items where Year is 2007 - UKM Journal Article Repository
Arifah Che Alhadi, and Lailatul Qadri Binti Zakaria, and **Shahrul Azman Mohd Noah**, and **Tengku Mohd Tengku Sembok**, (2007) Pengekstrakan dan perwakilan semantik dokumen web ...
journalarticle.ukm.my/view/year/2007.html - Cached

indirect co-occurrence

citation or present on same event

[PDF] 2011 International Conference on Semantic Technology and ...
Adobe PDF
Kasturi Dewi Varathan, **Tengku Mohd Tengku Sembok**, Rabiah Abdul Kadir and Nazlia ... Normaslina Jamil, Arifah Che Alhadi and **Shahrul Azman Noah** 119 Phonetic Coding ...
toc.proceedings.com/12500webtoc.pdf

DBLP: **Shahrul Azman Mohd. Noah** - DBLP Computer Science ...
Yamen Batch, Maryati Mohd Yusof, **Shahrul Azman Mohd. Noah**, Teh Phoey Lee: MTag: A ... Tengku M. T. Sembok (**Tengku Mohd Tengku Sembok**) 55: Tengku Mohd Tengku Sembuk
dblp.uni-trier.de/db/indices/a-tree/n/**Noah:Shahrul_Azman**... - Cached

Asia Information Retrieval Symposium 2010
Shahrul Azman Noah, Datul Aida Ali: The Role of Lexical Ontology in Expanding the ... Haibo Li, **Yutaka Matsuo**, Mitsuru Ishizuka: Semantic Relation Extraction Based on ...
www.**informatik.uni-trier.de**/~ley/db/conf/airs/airs2010.html - Cached

Hironori Tomobe - Microsoft Academic Search
Yutaka Matsuo, Hironori Tomobe, Takuichi Nishimura ... Hazalina Hashim, **Shahrul Azman Mohd Noah**
academic.research.microsoft.com/Author/1474459/hironori... - Cached

POLYPHONET: An advanced social network extraction system from ...
Yutaka Matsuo, Hironori Tomobe, Kôiti Hasida, Hideyuki Nakashima, Mitsuru ... Mahyuddin K M Nasution, **Shahrul Azman Noah** in 2011 International Conference on Semantic ...
www.**mendeley.com**/research/polyphonet-an-advanced-social... - Cached

Fig. 1. Type of snippets based on co-occurrence (Google search engine)

As information of the relations between social actors, the doubleton naturally be basic for refining the information about a social actor where one of search terms be a keyword for other. Therefore, this is sufficient condition for eliminating the connatural trait of the singleton. The snippets of doubleton, however naturally showed the different kind of information of relations. We conclude that based on snippets of doubleton there are the direct relations and the indirect relations, see Fig. 1. The snippet revealed an indirect relation with the presence of three (triple) dots between two names of social actors. Triple dots naturally is a word in text. The direct relations represented by direct co-occurrences like co-author, but the indirect relations represented by indirect co-occurrences such as citation or present on same event.

3 The Proposed Approach

The method of extracting information from Web recognized as the superficial method, categorized in unsupervised stream, involving a search engine to obtain the information like the hit counts used in computation [12]. Generally, for generating relation between actors applied the similarity measurement [13].

Definition 7. *$r_s \in R$ is the strength relation between two social actors $a, b \in A$ if it meets the comparison among the different information of two actors ($\mathbf{a}$ and $\mathbf{b}$) and the common information of them ($\mathbf{a} \cap \mathbf{b}$) in the similarity measurement. Or $sr = sim(\mathbf{a}, \mathbf{b}, \mathbf{a} \cap \mathbf{b})$ in $[0, 1]$, $\mathbf{a} \cap \mathbf{b} \leq \mathbf{a}$ and $\mathbf{a} \cap \mathbf{b} \leq \mathbf{b}$.*

Suppose we use Jaccard coefficient, we possess sr based on hit counts

$$sr = \frac{|\Omega_a \cap \Omega_b|}{|\Omega_a| + |\Omega_b| - |\Omega_a \cap \Omega_b|} \in [0, 1], \tag{5}$$

where $|\Omega_a \cap \Omega_b| \leq |\Omega_a|$ and $|\Omega_a \cap \Omega_b| \leq |\Omega_b|$.

Lemma 3. *If ir is a indirect relation between two social actors $a, b \in A$, then ir statistically has the character in the doubleton.*

Proof. Suppose the indirect relations ir can be recognized in each snippet based on doubleton, we have number of the indirect relations in the snippets list based on doubleton or $|ir|$, $|ir|$ = number of snippets contain triple dots. Therefore, we generate the character of ir as follows

$$p(ir) = \frac{|ir|}{|\Omega_a \cap \Omega_b|} \in [0, 1], \tag{6}$$

where $|ir| \leq |\Omega_a \cap \Omega_b|$ and $|\Omega_a \cap \Omega_b| \neq 0$.

Proposition 3. *If sr is a strength relation between two social actors $a, b \in A$, then the aggregation of sr consists of three binderies.*

Proof. Suppose $p(ir) \in [0, 1]$ (Eq. (6)) as probability of the indirect relation based on doubleton, then probability of the direct relation (dr) based on doubleton is as follows

$$p(dr) = \frac{|dr|}{|\Omega_a \cap \Omega_b|} \in [0, 1], \tag{7}$$

where $|dr| \leq |\Omega_a \cap \Omega_b|$ and $|\Omega_a \cap \Omega_b| \neq 0$.

Based on Lemma 3 and Eq. (7) we have two categories of relations based on doubleton, i.e. the indirect relation ir and the direct relation dr whereby their characteristics are $p(ir)$ and $p(dr)$, respectively. However, $1 - p(ir) - p(dr) \geq 0$, if $p(ir) + p(dr) - 1 \neq 0$, we obtain

$$p(ur) = 1 - (p(ir) + p(dr)) \tag{8}$$

i.e. the character of relation has not be determined with certainty through the co-occurrence. Because $p(ir)$, $p(dr)$ and $p(ur)$ can be considered as the percentage values, the multiplication of a characteristic with the strength relation regarded as bindery based on type of relations. Therefore, we have three bindings of the strength relations as follows

J1 A bindery of strength relations based on the direct relations,

$$sr_{dr} = sr * p(dr) \in [0, 1] \tag{9}$$

J2 A bindery of strength relations based on the indirect relations,

$$sr_{ir} = sr * p(ir) \in [0, 1] \tag{10}$$

J3 A bindery of strength relations based on the unclear relations,

$$sr_{ur} = sr * p(ur) \in [0, 1] \tag{11}$$

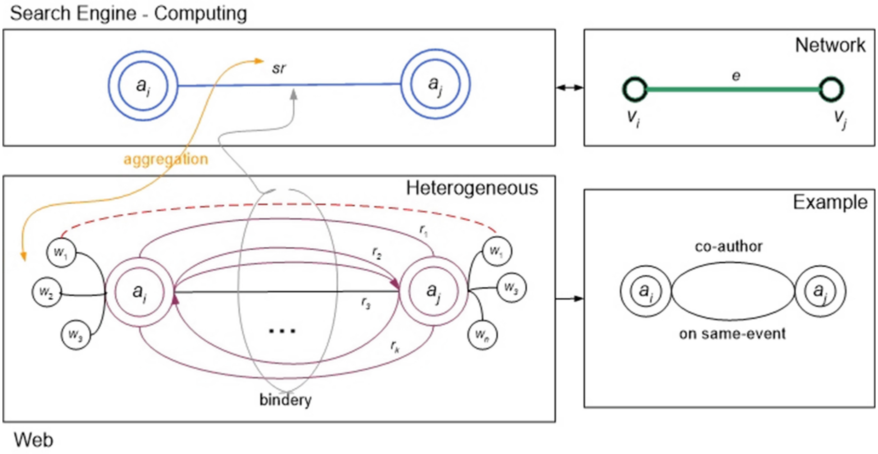


Fig. 2. Type of relations based the social network extraction

Proposition 4. *If sr is a strength relation between two social actors $a, b \in A$, then the aggregation of sr consists of sheets.*

Proof. Based on Proposition 3 and by applying Eq. (4) to the strength relation sr, we can generate the aggregations based on words and we call it as the sheets of relations sh, i.e.

$$sh = sr * p_{\mathcal{D}}(w) \in [0, 1] \tag{12}$$

Generally, this concept is considered to be an approach to the concept of latent semantic analysis [14] that have been put forward and produce labels on the social networks based on the supervised stream or the generative probabilistic model (PGM) [4,15]. This approach as enhancing for superficial method [16,17].

Theorem 1. *sr is the strength relation between two actors $a, b \in A$ if and only if there are aggregation.*

Proof. This is a direct consequence of Propositions 3 and 4 as the necessary conditions, and Lemmas 1, 2 and 3 as the sufficient conditions, see Fig. 2.

```
generate (keyword)
INPUT  : A set of actors
OUTPUT : aggregation of the strength relations
STEPS  :
```

1. $|\Omega_a| \leftarrow t_a$ query and search engine.
2. $|\Omega_b| \leftarrow t_b$ query and search engine.
3. $|\Omega_a \cap \Omega_a| \leftarrow t_a \wedge t_b$ query and search engine. $A = \{w_1, w_2, \ldots, w_n\} \leftarrow$ Collect words-(terms) per a pair of actors from snippets based on doubleton.
4. $|dr| \leftarrow$ List of snippets based on doubleton.
5. $|ir| \leftarrow$ List of snippets based on doubleton.
6. $sr * p(dr)$ and $sr * p(ir)$
7. Aggregating $sr * p(dr)$ and $sr * p(ir)$ based on the summation of sheets per domain.
8. Measuring recall and precision of relations.

4 Experiment

In this experiment, we implicate $n = 469$ social actors or $n(n-1) = 219,492$ potential relations. There are 30,044 strength relations between 469 actors or 14% of potential relations, among them (a) 4,422 direct relations (2%), (b) 21,462 indirect relations (10%), and (c) 4,160 direct and indirect relations (2%). Therefore, there are 21,462 lists of snippets of doubleton ($\mathcal{L}_{\mathcal{D}}$) contain the triple dots in all snippets, or there are 4,422 lists of snippets of doubleton ($\mathcal{L}_{\mathcal{D}}$) have no dots in all snippets.

Suppose we define the ontology domain and taxonomically we interpret in a set of words as follows

1. Direct relations:
   (a) author-relationship = {activity, article, author, authors, award, journal, journals, paper, patent, presentation, proceedings, publication, theme, poster, ...}.
   (b) academic rule = {supervisor, cosupervisor, editor, editors, graduate, lecturer, professor, prof, researcher, reviewer, student, ...}.
   (c) research group = {association, committee, group, institute, lab, laboratory, member, team, project, ...}.

Table 1. The strength relation, direct and indirect relations, and author-relationship

	sr	
1. Abdullah Mohd Zin	0.0482	0.0395
2. Abdul Razak Hamdan		0.0237
3. Tengku Mohd Tengku Sembok		

	dr	*ir*	*dr*	*ir*
1. Abdullah Mohd Zin	0.0163	0.0815	0.0975	0.0612
2. Abdul Razak Hamdan			0.0000	0.2349
3. Tengku Mohd Tengku Sembok				

	1 & 2	1 & 3	2 & 3
activity	0.0023	0.0027	0.0000
article	0.0258	0.0613	0.0000
author	0.0103	0.1465	0.0000
authors	0.0156	0.0649	0.0001
journal	0.0371	0.1316	0.0001
journals	0.0400	0.0043	0.0000
paper	0.0289	0.0725	0.0000
patent	0.0000	0.0025	0.0000
presentation	0.0083	0.0299	0.0000
preceedings	0.0118	0.0497	0.0001
publication	0.0020	0.0115	0.0000

2. Indirect relations:
   (a) scientific event = {chair, conference, conferences, meeting, programme, schedule, seminar, session, sponsor, symposium, track, workshop, . . . }.
   (b) citation = {reference, references, bibliography, . . . }

With the concept of aggregation starting from the bindery, each bindery consists of chapters (domains), and each chapter contains the sheets (words).

For example, hit counts ($|\Omega_a|$) of “Abdullah Mohd Zin”, “Abdul Razak Hamdan”, and “Tengku Mohd Tengku Sembok” are 7,740, 8,280, and 3,860, respectively. While $|\Omega_a \cap \Omega_b|$ between “Abdullah Mohd Zin” and “Abdul Razak Hamdan” is 736, $|\Omega_a \cap \Omega_c|$ between “Abdullah Mohd Zin” and “Tengku Mohd Tengku Sembok” is 441, and $|\Omega_b \cap \Omega_c|$ between “Abdullah Razak Hamdan” and “Tengku Mohd Tengku Sembok” is 281. Therefore, based on Eq. (5) we have three strength relations *sr* like Table 1. From 100 snippets based on doubleton, we have:

1. 60 snippets contain the indirect relations and 12 snippets contain the direct relations for “Abdullah Mohd Zin” and “Abdul Razak Hamdan”,
2. 27 snippets contain the indirect relations and 43 snippets contain the direct relations for “Abdullah Mohd Zin” and “Tengku Mohd Tengku Sembok”, and
3. 66 snippets contain the indirect relations for “Abdul Razak Hamdan” and “Tengku Mohd Tengku Sembok”.

In this case, $p(dr)$ and $p(ir)$ for a pair of actors there are in Table 1. While 100 snippets for each pair of actors are calculated $p_{\mathcal{D}}(w)$ for each word and its value is directly transferred to the sheets in the appropriate domain, such as Table 1.

Table 2. .

	Aggregation	Recall	Precision
1	Author-relationship	61.76%	17.65%
2	Research group	55.88%	7.28%
3	Academic rule	61.94%	13.15%
4	Scientific event	61.76%	6.10%
5	Citation	50.01%	6.63%

We conduct an experiment using 65 social actors that have direct and indirect relations between them, or $n(n-1) = 4,160$ potential relations. Based on survey we obtain the relevant relation and this is a comparison of the results obtained through extraction from Web. Based on Table 2, the recall and the precision give the impression that the activation of each aggregation of the strength relation as adequate.

5 Conclusion and Future Work

By studying the principle of methods for extraction the relation between social actors, we have an enhanced method for aggregation the relations to interpret more rich about social. Thus, this new method still needs further verification. Future work we study about combination between sheets and domain based on ontology.

References

1. Adamic, L.A., Adar, A.: Friends and neighbours on the web. Soc. Netw. **25**, 211–230 (2003)
2. Nasution, M.K.M., Noah, S.A.: Superficial method for extracting social network for academics using web snippets. In: Yu, J., Greco, S., Lingras, P., Wang, G., Skowron, A. (eds.) RSKT 2010. LNCS (LNAI), vol. 6401, pp. 483–490. Springer, Heidelberg (2010). doi:10.1007/978-3-642-16248-0_68
3. Cullota, A., Bekkerman, R., McCallum, A.: Extracting social networks and contact information from email and the Web. In: Proceedings of the 1st Conference on Email and Anti-Spam (CEAS) (2004)
4. McCallum, A., Corrada-Emmanual, A., Wang, X.: The author-recipient-topic model for topic and role discovery in social networks, with application to Enron and academic email. In: Proceedings of the Workshop and Link Analysis, Counterrorism, and Security (in Conjunction with: SIAM International Conference on Data Mining), pp. 33–44 (2005)

5. Heras, S., Atkinson, K., Botti, V., Grasso, F., Julián, V., McBurney, P.: Research opportunities for argumentation in social networks. Artif. Intell. Rev. **39**, 39–62 (2013)
6. Kautz, H., Selman, B., Shah, M.: ReferralWeb: combining social networks and collaborative filtering. Commun. ACM **40**(3), 63–65 (1997)
7. Finin, T., Ding, L., Zhou, L., Joshi, A.: Social networking on the semantic web. Learn. Organ. **12**(5), 418–435 (2005)
8. Nasution, M.K.M., Sitompul, O.S., Sinulingga, E.P., Noah, S.A.: An extracted social network mining. In: SAI Computing Conference. IEEE (2016)
9. Nasution, M.K.M.: Social network mining (SNM): a definition of relation between the resources and SNA. Int. J. Adv. Sci. Eng. Inf. Technol. **6**(6), 975–981 (2016)
10. Nasution, M.K.M.: Modelling and simulation of search engine. In: International Conference on Computing and Applied Informatics (ICCAI). IOP (2016)
11. Nasution, M.K.M.: New method for extracting keyword for the social actor. In: Nguyen, N.T., Attachoo, B., Trawiński, B., Somboonviwat, K. (eds.) ACIIDS 2014. LNCS (LNAI), vol. 8397, pp. 83–92. Springer, Cham (2014). doi:10.1007/978-3-319-05476-6_9
12. Matsuo, Y., Mori, J., Hamasaki, M., Nishimura, T., Takeda, T., Hasida, K., Ishizuka, M.: POLYPHONET: an advanced social networks extraction system from the web. J. Web Semant. Sci. Serv. Agents World Wide Web **5**, 262–278 (2007)
13. Nasution, M.K.M.: New similarity. In: Annual Applied Science and Engineering Conference (AASEC). IOP (2016)
14. Blei, D.M., Ng, A.Y., Jordan, M.J.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
15. McCallum, A., Corrada-Emmanual, A., Wang, X.: Topic and role discovery in social networks. In: Proceedings of the 19th International Joint Conference on Artificial Intelligence, pp. 786–791 (2005)
16. Nasution, M.K.M., Mohd Noah, S.A.: Extraction of academic social network from online database. In: Mohd Noah, S.A. et al. (eds.) Proceeding of 2011 International Conference on Semantic Technology and Information Retrieval (STAIRS 2011), pp. 64–69. IEEE, Putrajaya (2011)
17. Nasution, M.K.M., Noah, S.A.: A social network extraction perspective. In: CAMP 2012. IEEE (2012)

A Case Study on Flat Rate Estimation Based on Fuzzy Logic

Kazi Kowshin Raihana, Fayezah Anjum,
Abu Saleh Mohammed Shoaib, Md. Abdullah Ibne Hossain,
M. Alimuzzaman, and Rashedur M. Rahman(✉)

Department of Electrical and Computer Engineering, North South University,
Plot 15, Block-B, Bashundhara, Dhaka 1229, Bangladesh
kowshin.raihana@gmail.com, fayezah16@gmail.com,
saleh.shoaib@northsouth.edu, irfanhossain23@yahoo.com
alimuzzamanprince@gmail.com,
rashedur.rahman@northsouth.edu

Abstract. The main objective of this research is to develop a system based on fuzzy logic to estimate flat rent in Bangladesh. The data set consists of 63 individual flats in different blocks of Bashundhara Residential Area, Dhaka, Bangladesh. A number of factors influence the decision of a tenant to rent a flat. Since it is not desirable to work with a large number of variables, we used one of the dimensionality reduction techniques, known as Principle Component Analysis (PCA). PCA helps in reducing the number of variables from the set of data, as well keeping hold the most of the variability in data. This paper describes the implementation of an adaptive neuro-fuzzy inference system (ANFIS)-based approach to estimate the rent of the flat. The Sugeno ANFIS model is proposed in order to develop a systematized approach of generating fuzzy rules and membership function parameters for fuzzy sets from a given set of input and output data.

Keywords: Principle Component Analysis (PCA) · Adaptive Neuro-Fuzzy Inference System (ANFIS) · Fuzzy logic · Rent estimation system · Scatter plot · Sugeno type model

1 Introduction

Fuzzy logic is a computing approach that allows the value of the variable in the interval [0,1] whereas in the Crisp logic the value can be either 0 or 1. In 1965, Lotfi A Zadah introduced fuzzy sets where a more flexible use of membership is possible. In fuzzy sets, many degrees of memberships are allowed. To control and model uncertain system in industry or for any real life applications fuzzy logic is a powerful mathematical tool.

The main goal of this paper is to develop a system based on fuzzy logic where the flat rent of any apartment in Bangladesh can be estimated. In this paper, the variables used for data analysis are collected from flats in different blocks of Bashundhara

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_32

Residential Area, Dhaka, Bangladesh. The data is collected through surveys, from which we could also determine different preferences.

The rent of a flat is dependent on many variables. In our survey we have included 14 variables; 5 out of them are crisp, 5 are fuzzy and the remaining 4 variables can be answered by yes/no. Then Principal component analysis (PCA) is applied to reduce the number of inputs and then fed to the ANFIS model. PCA is a dimensionality reduction technique that reduces the number of inputs. It is especially useful for simplifying the interpretation of highly multivariate data sets by reducing the number of variables [4]. It still captures most of the variability of the original data set.

The output from ANFIS predicts the estimated rent. The Sugeno ANFIS model presented in this paper was proposed to develop a systematic approach for generating fuzzy rules and membership function parameters for fuzzy sets from a given input–output data set [1]. There are few papers based on real estate valuation using fuzzy logic, but no direct work has been done for flat rent estimation system. Since this is a new topic and no work has been done based on this topic in Bangladesh so far, we had some constraints to deal with. There are number of scopes for future development that we report at the end of the paper.

2 Related Works

The authors in [5] have proposed an ANFIS based approach for assessment of real estate property. The data used in the paper consists of values from the sales of houses in a market in the Midwest region of the United States. Guan, Zurada and Levitan [5] exhibited the first attempt to take account of the application of ANFIS in valuing real estate properties. The results obtained from the research were compared and analysed with the outcome a traditional multiple regression model. From that study, it was clearly shown that ANFIS was a feasible approach in assessment of real estate property values.

The authors of paper [6] described the fuzzy expert system for real estate recommendation. In the paper, the architecture and design of the system was also proposed by the authors. Four essential parameters such as price, region, plan and facilities that are needed for real estate recommendation were taken into account, and these variables were obtained through the distribution of questionnaires among experts of real estate. The system was then compared with the conventional one, and at the end, it was proved that the proposed system in this study was less time consuming and incorporated all the functions as expected by the designers.

3 Data Set Description

We used 14 variables in our research. Those are discussed below:

(1) **Distance from Main Road**: The far the apartment from the main road, the lower the rent of the flat is.

(2) **Size of the flat:** It is needless to say that the flat size has direct influence on the flat rent.

(3) **Rent:** It is needed for generating estimation rules which is the main objective of the research.

(4) **Floor:** People give preference in the location of floor. The higher the floor is located, the lower the price is. Generally it is noticed that the 2nd floor is the most preferable.

(5) **Number of Bedrooms and Bathrooms:** It is also important variable that influence the rent. It also dependents on personal preferences.

(6) **Transportation Availability:** The higher the transportation availability, the more preferable the flat is.

(7) **Lift, Garage, Generator:** Availability of these facilities definitely causes increase in the flat rent.

(8) **Security Level:** The higher the security level is, the more preference the flat gets regardless the type of area or tenant.

(9) **Road condition, Water & Gas supply, Ventilation & Light:** These are some environmental parameters that have more or less effect on a particular flat rent.

(10) **Furnished by Flat Owner:** This variable has an effect since full furnished flats can be preferable to students who rent a flat until their end of academic year.

The data set consists of 63 individual flats in different blocks of Bashundhara residential area, Bangladesh. From the total of 63 data sets, 27 were collected through door-to-door survey and remaining 36 were collected using online survey questionnaire. Initially 52 online surveys were filled up, but due to several errors such as inconsistent information, blank answer space and so on, 16 data sets were excluded. One sample data set is given in Table 1.

Table 1. Sample data set

Variables	Sample data after transformation
Distance from main road (km)	1.8
Size (Square ft.)	2100
Rent (Tk) (Output)	30,000
Floor	5
Number of bedrooms	3
Number of bathrooms	4
Transportation availability	8
Lift	1
Furnished by the landlord	1
Road condition	9
Security	8
Garage	1
Generator	1
Water & gas supply	5
Ventilation & light	8

It could be seen from Table 1 that distance from main road, size, floor, number of bedrooms and bathrooms are five variables that can take crisp inputs. From the remaining variables, transportation availability, road condition, security, water & gas supply and ventilation and light are fuzzy variables. Finally, the availability of the facilities of lift, furnished by the landlord, garage and generator can be answered only by saying ‘YES’ or ‘NO’.

4 Methodology

Our rent estimation system has two main parts which are PCA and ANFIS. For better understanding the architecture is given in Fig. 1.

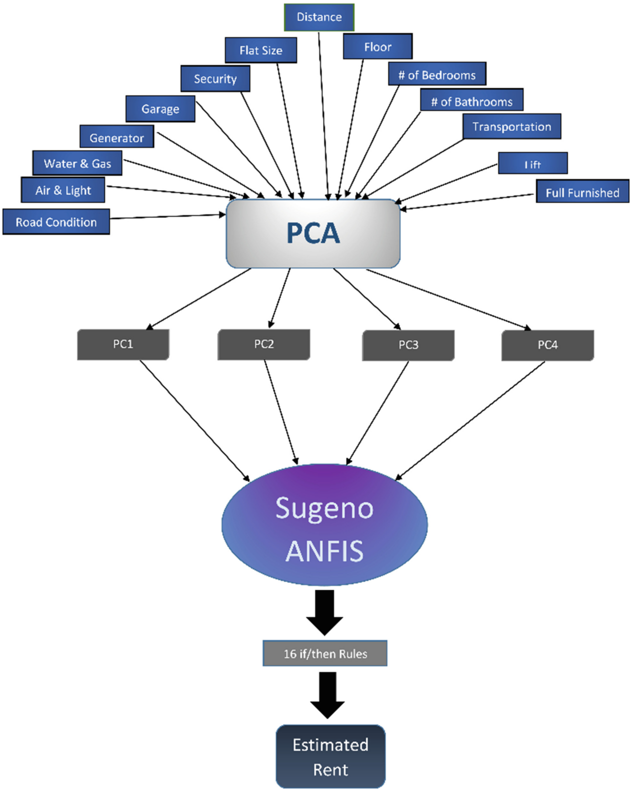


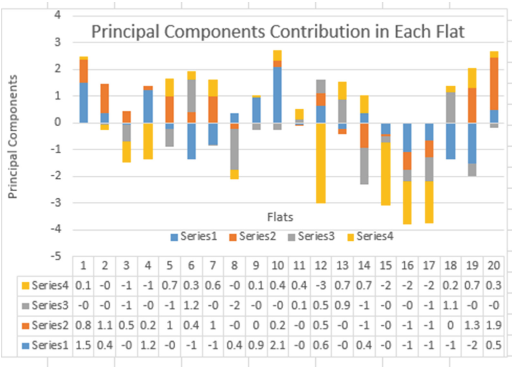
Fig. 1. Rent estimation system architecture

PCA: For Principal Component Analysis, the values of all variables need to be in crisp form. The 5 variables with crisp value were taken as they are, the 5 fuzzy variables values are taken as a rating from 1 to 10, where 1 is worst and 10 is best. For the remaining 4 variables, the taken value is 0 if the answer is ‘NO’ and 1 if the answer is

‘YES’. The sample data input is given in Table 1 for each case. The 14 inputs are reduced to 4 principal components using princomp(data matrix) function in MATLAB R2016b.

The PCA analysis on our data matrix generates the coefficients matrix, i.e., the matrix of data values transformed into the principal component space and the vector containing the eigenvalues [3]. From the generated eigenvalues, it is observed that the 1st four principal components capture 99.9% of the total variability in the data set [5].

Figure 2 shows the contribution of the 4 principal components in the data set where series1, series2, series3 and series4 represent PC1, PC2, PC3 and PC4 respectively. In horizontal axis 20 flats of the 63 are represented. There are 20 stacked columns along the vertical axis that are the Principal component contributions of PC1(series 1), PC2 (series 2), PC3(series 3) and PC4(series 4).



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Series4	0.1	-0	-1	-1	0.7	0.3	0.6	-0	0.1	0.4	0.4	-3	0.7	0.7	-2	-2	-2	0.2	0.7	0.3
Series3	-0	-0	-1	-0	-1	1.2	-0	-2	-0	-0	0.1	0.5	0.9	-1	-0	-0	-1	1.1	-0	-0
Series2	0.8	1.1	0.5	0.2	1	0.4	1	-0	0	0.2	-0	0.5	-0	-1	-0	-1	-1	0	1.3	1.9
Series1	1.5	0.4	-0	1.2	-0	-1	-1	0.4	0.9	2.1	-0	0.6	-0	0.4	-0	-1	-1	-1	-2	0.5

Fig. 2. Stacked column s of principal components in each flat

The outputs of the PCA are the 4 Principal Components for each of the 63 cases. One sample output is shown in Table 2.

Table 2. Sample output of PCA

Principal components	Values
PC1	1.225080245
PC2	0.155955533
PC3	−0.027694021
PC4	−1.346951764

ANFIS: The Sugeno ANFIS model is selected for estimating an approximation of flat rent. Using PCA, now the input of this ANFIS has become the 4 Principal Components PC1, PC2, PC3 and PC4 and the corresponding flat rent will be the output. After training the data and generating FIS, the structure of the ANFIS is as shown in Fig. 3.

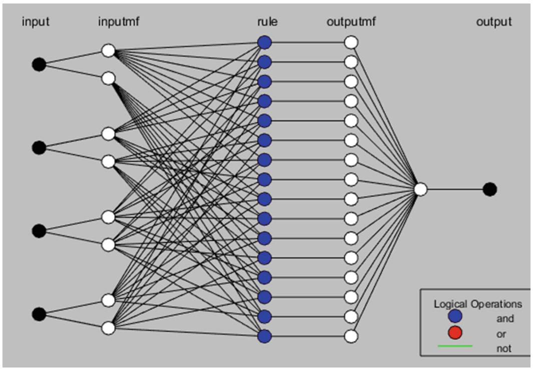


Fig. 3. Generated ANFIS structure

Each Principal Component has two Gaussian membership functions for the two linguistic labels SMALL and LARGE as shown in Table 3.

Table 3. Gaussian membership functions of the principal components

PC	Linguistic label	σ	c	Membership function (μ)
PC1	SMALL	1.919	−1.497	$e^{-\frac{(x+1.497)}{7.3651}}$
	LARGE	1.638	2.074	$e^{-\frac{(x-2.074)}{5.3661}}$
PC2	SMALL	1.948	−2.599	$e^{-\frac{(x+2.599)}{7.5894}}$
	LARGE	1.95	1.91	$e^{-\frac{(x-1.91)}{7.605}}$
PC3	SMALL	2.54	−2.417	$e^{-\frac{(x+2.417)}{12.9032}}$
	LARGE	2.45	3.48	$e^{-\frac{(x-3.48)}{12.005}}$
PC4	SMALL	2.576	−2.974	$e^{-\frac{(x+2.974)}{13.2716}}$
	LARGE	2.519	3.172	$e^{-\frac{(x-3.172)}{12.6907}}$

The eight membership functions of the 4 input Principal Components produce 16 if-then rules in total that are shown in Fig. 4.

The generated rules by Sugeno type ANFIS are given in Table 4. Here each variable values are fuzzy numbers with different ranges that correspond to different grade of memberships.

The generated surface graphs to show the relation between PC1, PC2, PC3 and PC4 with the output of estimated rent are shown in Fig. 5.

Surface graphs are shown in Fig. 5. Figure 5(a) depicts PC1 vs. PC2 vs. estimated rent, Fig. 5(b) depicts PC2 vs. PC4 vs. estimated rent, Fig. 5(c) depicts PC2 vs. PC3 vs. estimated rent in and in Fig. 5(d) PC3 vs. PC4 vs. estimated rent is shown.

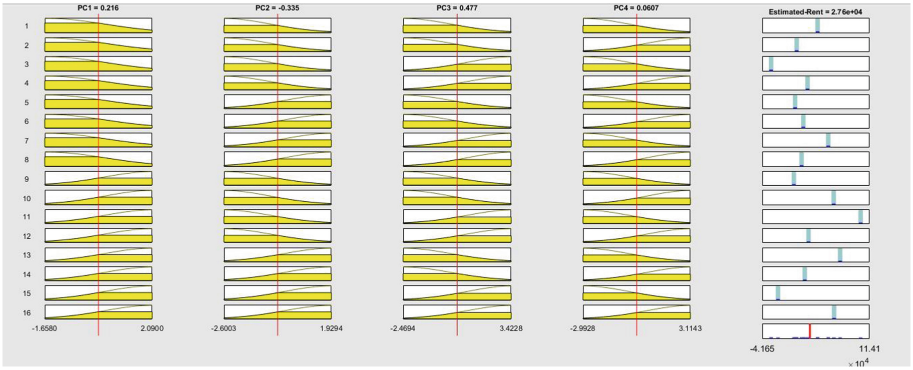


Fig. 4. The generated rule view of the ANFIS

Table 4. Rules for rent estimation.

Rule no.	PC1	PC2	PC3	PC4	Estimated rent(Around) (tk)
1.	SMALL	SMALL	SMALL	SMALL	38750
2.	SMALL	SMALL	SMALL	LARGE	8438
3.	SMALL	SMALL	LARGE	SMALL	28200
4.	SMALL	SMALL	LARGE	LARGE	24120
5.	SMALL	LARGE	SMALL	SMALL	6114
6.	SMALL	LARGE	SMALL	LARGE	18000
7.	SMALL	LARGE	LARGE	SMALL	54170
8.	SMALL	LARGE	LARGE	LARGE	15510
9.	LARGE	SMALL	SMALL	SMALL	4267
10.	LARGE	SMALL	SMALL	LARGE	62090
11.	LARGE	SMALL	LARGE	SMALL	101200
12.	LARGE	SMALL	LARGE	LARGE	25540
13.	LARGE	LARGE	SMALL	SMALL	71390
14.	LARGE	LARGE	SMALL	LARGE	20140
15.	LARGE	LARGE	LARGE	SMALL	31200
16.	LARGE	LARGE	LARGE	LARGE	62620

5 Result Analysis

We divide the 63 cases into two groups. One group contains 49 cases and another contains 14 cases. We perform PCA on both groups separately and get 4 PCs in both cases. We use the PCA outputs of 49 data for training purpose, and the PCA outputs of 14 cases for the testing purpose. We use 5 membership functions and 40 epochs in training case.

It can be seen from Fig. 6(a) that the plotting of the FIS output accurately matches with the plotting of the training data. In this case, the average training error is 0.50542.

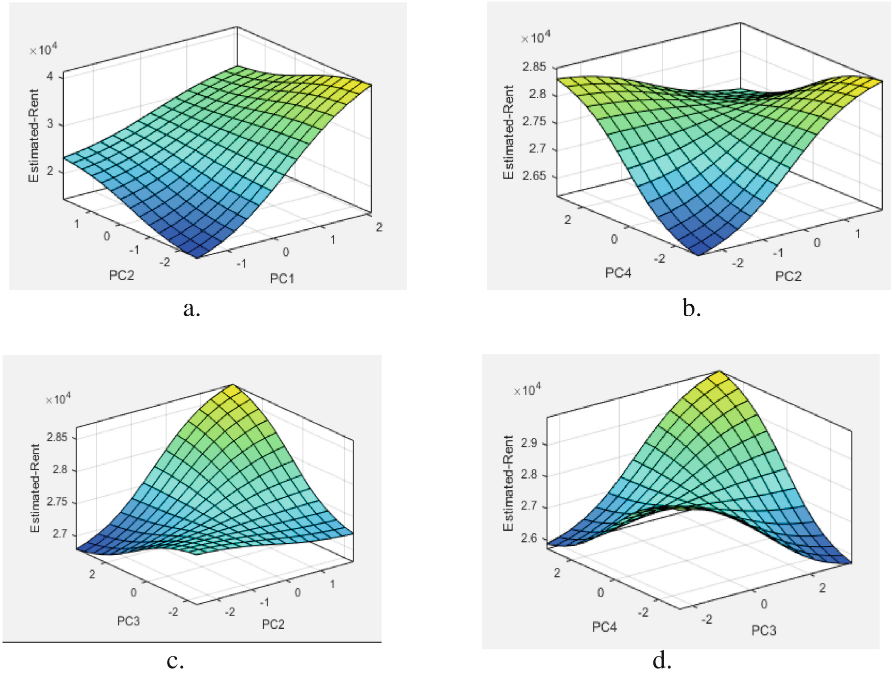


Fig. 5. Surface graphs

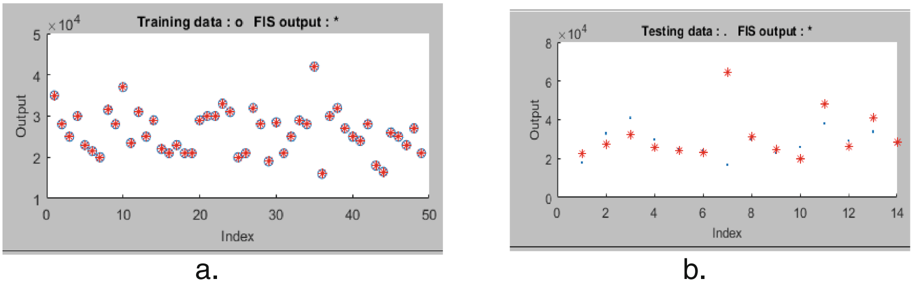


Fig. 6. FIS output plotted against training data (a) and testing data (b).

Though training data plotting matched with the FIS output with a small average error but from Fig. 6(b) it is seen that the average testing error is comparatively high. The reason behind this is the shortage of testing data. As we manage to collect only 63 data sets, the 77% of data has been used as training set and remaining 22% data is used as testing set. The data set of the 14 flats hardly justify the range of the training data to begin with [2].

From the 14 testing data set we took some cases to compare their actual rent with the ANFIS generated rents. Table 5 depicts this.

Table 5. Comparison of rents of random 5 test cases

Test case no.	PC1	PC2	PC3	PC4	Source	Rent
1.	0.131	−1.24	−0.424	−0.664	ANFIS	28,200
					ACTUAL	28,000
2.	−0.7	0.105	0.795	0.759	ANFIS	25,100
					ACTUAL	25,000
3.	0.522	−1.66	−0.912	0.0732	ANFIS	27,600
					ACTUAL	30,000
4.	−0.211	0.227	1.77	−2.13	ANFIS	22,300
					ACTUAL	24,500
5.	−0.945	0.656	0.709	1.13	ANFIS	25,400
					ACTUAL	23,000

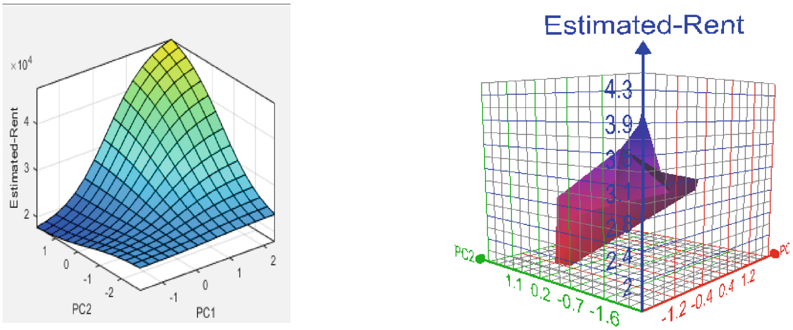


Fig. 7. ANFIS generated smooth surface view (left) and Graphing Calculator 3D generated rough surface graph (right) for PC1 vs. PC2 vs. estimated rent

From Table 5, we see that case 2 contains PC1, PC2, PC3 and PC4 values of −0.700425108, 0.105194481, 0.795480446 and 0.758559801 respectively with rent of 25,000tk. Approximate values of the PC1, PC2, PC3 and PC4 were inputted in the rule view of the ANFIS. As it can be seen from Table 5, ANFIS outputs nearly the same value of 25,100tk. Therefore, from Table 5 it can be seen that the ANFIS generated estimated rents are fair predictions of the actual rents.

We conduct a surface graph shape test. We generate the surface view from PCA outputs from the ANFIS of 49 training data sets. Then we use the PCA outputs of the remaining 14 data sets to directly plot a surface graph using the software Graphing Calculator 3D separately. The axis ranges were kept same to compare the surface graphs on both cases.

PC2, PC3, and PC4 is plotted along Y axis in Figs. 7, 8 and 9 respectively and along X-axis PC1 is plotted in all figures. Estimated Rent is in the Z-axis. It can be seen that there is a good similarity between the two graphs in shapes. The small data set (14 flats) generated surface looks like a small part of the large data (49 flats) set generated surface. The dissimilarities could have been decreased if the data sets could be made of

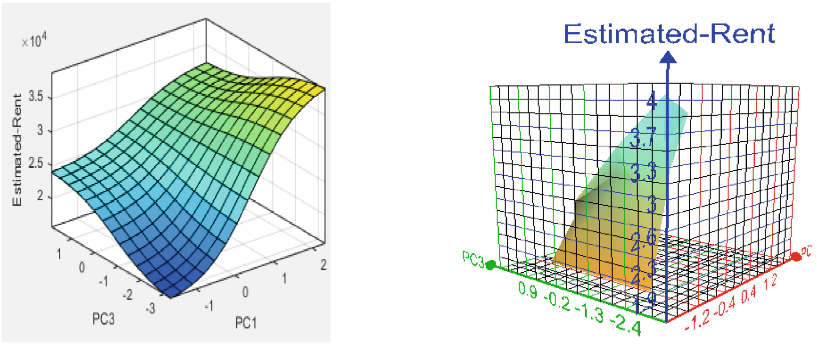


Fig. 8. ANFIS generated smooth surface view (left) and Graphing Calculator 3D generated rough surface graph (right) for PC1 vs. PC3 vs. estimated rent

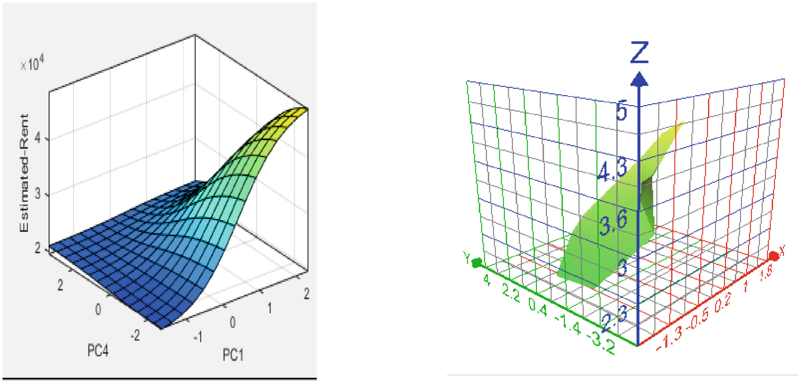


Fig. 9. ANFIS generated smooth surface view (left) and Graphing Calculator 3D generated rough surface graph (right) for PC1 vs. PC4 vs. estimated rent

the same size. This proves that the ANFIS generated results reflect similarity with the practical scenario.

6 Future Work

There are very few researches based on real estate valuation using fuzzy logic, but no direct work has been done for flat rent estimation system. Since this is a new topic and no work has been done based on this topic in Bangladesh so far, we had some constraints due to which we could not develop a fully functional system.

There is lot of scopes for future development. As a first attempt we have developed a system based on fuzzy logic to estimate the flat rent in Bangladesh from a small data set based on the flats of Bashundhara area. In future, we plan to create a database where all the data regarding different variables and user choice will be stored for easy access.

We also plan to develop software which would help in graphical demonstration of the desired flat with their corresponding size and estimated rent.

7 Conclusion

The creation of membership functions and fuzzy rules are one of the major challenges in implementation of fuzzy logic. In case of simple applications, it is feasible to construct these using common sense or domain knowledge, but in complicated system where there is huge volume of data set, it is difficult to predict. This paper represents the initial attempt to create a system to estimate flat rent based on Sugeno model.

This research paper is focused on finding out estimated rent for flats in Bangladesh. By using 3 types result analysis, it is shown that the process is factual and the results are nearly similar. This paper found accuracy while observing average training error, comparing ANFIS generated rent and actual rent. There is also similarity in surface graph generated from training cases and rough graph generated from testing cases. As the result is based on small number of data, therefore, by adding more data set the result will be more convincing.

There are many limitations in this study. Since it is a new research area, we had no reference to past works. There is no standard or sample data available; we had to rely on surveys to get the user preference. Also due to time constraint it was not possible for us to develop the complete commercial software. These limitations indicate the possible future works that can be done based on the research outcome from this project.

References

1. Sugeno, M., Kang, G.T.: Structure identification of fuzzy model. Fuzzy Sets Syst. **28**, 15–33 (1988)
2. Giudici, P.: Applied Data Mining: Statistical Methods for Business and Industry. Wiley, Chichester (2003)
3. Principal component analysis (PCA) on data - MATLAB princomp (2017). https://www.mathworks.com/help/stats/princomp.html. Accessed 10 Jan 2017
4. Stepnowski, A., Mosynski, M., Dung, T.V.: Adaptive neuro-fuzzy and fuzzy decision tree classifiers as applied seafloor characterization. Acoust. Phys. **49**(2), 189–192 (2003)
5. Guan, J., Zurada, J., Levitan, A.S.: An adaptive neuro-fuzzy inference system-based approach to real estate property assessment. J. Real Estate Res. (JRER) **30**(4), 396–421 (2008). pages.jh.edu/jrer/papers/pdf/past/vol30n04/01.395_422.pdf
6. Kafi, A.H., Kazemipoor, H., Afshar Kazem, A.M.: Design and implementation of fuzzy expert system for real estate recommendation. Int. J. Inf. Secur. Syst. Manag. (IJISSM) **2**(1), 142–147 (2013). http://www.ijissm.org/article_7443.html

SA-Optimization Based Decision Support Model for Determining Fast-Food Restaurant Location

Ditdit Nugeraha Utama(✉), Muhammad Ryanda Putra, Mawaddatus Su'udah, Zulfah Melinda, Nur Cholis, and Abdul Piqri

Laboratory Optimization Models and Systems for Decision Support,
Information Systems Department, UIN Syarif Hidayatullah, Jakarta, Indonesia
dit.utama@uinjkt.ac.id

Abstract. An intense study is required to place a fast-food restaurant strategically. Several aspects should be taken into account to locate the restaurant in the right location to give owners high profit or gain; e.g. population income, city type, trading activity behavior, location distance to hustle in the city, etc. In addition, the simulated annealing (SA) was used as a main method for optimizing process. The other method fuzzy-logic was operated as well to define the selected parameters' priority based on experts' justification. The decision support model (DSM) based on SA-optimization and fuzzy-concept was constructed then. It practically proposed the best location alternative coming from a lot of alternatives. Here, three locations in Tangerang and Jakarta were undertaken as a research object.

1 Introduction

One of the most essential circumstances for a restaurant success is its location (Tzeng et al. 2002). Specifically concerning fast-food restaurant, one its characteristic is inexpensive price (Kim and Leigh 2011). It will give a lower profit than other types. Thus, the well-panned location will be valuable to encourage the restaurant operation through taking gain of it.

To place the location of fast-food restaurant is not easy way to do. The deep study should be conducted to do so by considering various parameters. The model to decide the decision to choose the right location is realistically necessitated then. Here, the model can support the decision maker to make the decision objectively.

Such a model has been studied and proposed in several fields by many researchers. Glock (2017) has systematically reviewed the decision support model that correlates closed-loop supply chain management involving returnable transport item. Pattanaik and Yadav (2015) developed a DSM for automated railway level crossing. They used fuzzy logic control as a main method of the model. The model provides intelligent decisive action signals as similar to a human brain. Also Piltan and Sowlati (2016) constructed a DSM. They used models interpretive structural modeling, analytical network process, and fuzzy logic to address the interdependency in evaluating the performance of partnerships. This model

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_33

was applied in partnership between a logging and sawmill companies in British Columbia, Canada.

In addition, Syam and Bhatnagar (2015) proposed a DSM to determine product variety for a manufacturer. They conducted it by seeing marketing supply chain perspectives. The simulation examined the optimal level of product variety. Hong et al. (2012) developed DSM for reducing electric energy consumption of facilities that was implemented in an elementary school. The data used to describe the characteristic of electric energy consumption were coming from more than 6,000 elementary schools in seven metropolitan cities in South Korea. And Rosenfeld and Shohet (1999) developed a DSM for automatically selecting a facility renovation alternative. The model proposed reasonable and economic policy of rehabilitating, renovating, remodeling, or rebuilding the facility.

In here, the paper talks the constructed DSM that exclusively can support to locate the fast-food restaurant in objective location. The introduction part will be followed by following parts; research methodology, results and discussions, and also conclusion and further works. For the study purpose, three areas in Tangerang Selatan and Jakarta Selatan (Indonesia) are taken as a research object.

2 Related Works

Particularly, various researches regarding restaurant location case have been conducted by many researchers. Sloan et al. (2016) have conducted research concerning the relationship between entrepreneurship and crime. They concluded that each crime is positively related to the number of new restaurants. Thornton et al. (2016) investigated the locations of fast-food restaurant across the state of Victoria (Australia) relative to area-level disadvantage, urban-regional locality, and around schools. Research findings exposed greater locational access to fast food restaurants in more socioeconomically disadvantaged areas, nearby to secondary schools, and nearby to primary and secondary schools within the most disadvantaged areas of the major city region. Chen and Tsai (2016) developed a data mining framework to support location selection decisions. This study focused on a restaurant chain to demonstrate the validity of the proposed framework. And, in geography field, Dock et al. (2015) also tried to construct a model to evaluate the restaurant location and competitiveness in Jefferson County (Kentucky).

Furthermore, Tzeng et al. (2002) have proposed a method for decision making based on multi criteria to select a restaurant location. This research was conducted in Taipei. They used five aspects and eleven criteria to develop a restaurant location evaluation hierarchy. Min (1987) also proposed a multi objective decision model for solving the problem regarding fast-food restaurant location. In developing the model, they considered behavioral and spatial aspects of location scenario, and also took advantages of systematic decision process.

3 Research Methodology

Principally, there are three main objectives of the research, where they were reached through following three steps with several methods operated. The framework of the research is clearly represented in Fig. 1. To define the problem and decision parameters, two methods desk based research and literature study were used. This objectives were gotten by rationally following the research step "initial and situational analyzing".

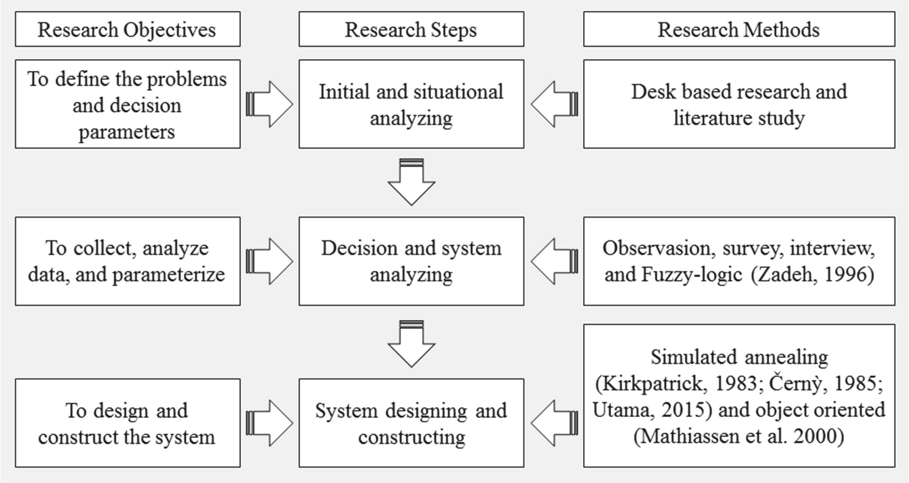

Research Objectives
Research Steps
Research Methods
To define the problems and decision parameters
Initial and situational analyzing
Desk based research and literature study
To collect, analyze data, and parameterize
Decision and system analyzing
Observasion, survey, interview, and Fuzzy-logic (Zadeh, 1996)
To design and construct the system
System designing and constructing
Simulated annealing (Kirkpatrick, 1983; Černỳ, 1985; Utama, 2015) and object oriented (Mathiassen et al. 2000)

Fig. 1. Research framework

Step "decision and system analyzing" fulfilled three types of research objectives: to collect and analyze data, and parameterize. For supporting the step, three areas in Tangerang Selatan and Jakarta Selatan (Indonesia) were used as a research object. That three types of empirical data were functioned as a basic data behavior to generate more than 1,000 experimental data for purpose of model simulation. In this step, based on Kunsoon (2002), fifteen decision parameters that grouped into six categories were defined as well. The urgency of decision parameters were technically justified by two experts and converted to become coefficient values via the concept of fuzzy logic (Zadeh 1996).

Simulated annealing (Kirkpatrick et al. 1983; Černý 1985; Utama 2015) was benefitted as a main method of optimization. The optimization model was used to propose a location as the most objective decision where the fast-food restaurant would be placed in. The constructed DSM also was developed through method object oriented (Mathiassen et al. 2000); where several types of unified modelling language (UML) notation were technologically employed, e.g. usecase diagram, class diagram, etc.

4 Results and Discussions

4.1 Constructed Model

Fifteen decision parameters operated in the model are described in the Table 1, where they are grouped into six categories: general location, position of site, demographics, traffic information, competition, and cost consideration. They were justified by two experts to see their urgency level. The urgency levels are produced through mechanism of fuzzy logic concept, they are represented via normal coefficient value (can be seen in column coefficient value). The membership function of fuzzy itself that was used in fuzzy-de-fuzzy process is described in Fig. 2. It is in fuzzy triangular membership function form with five fuzzy languages (very unimportant, unimportant, rather important, important, and very important).

In addition, the usecase diagram of the constructed model is visibly depicted in Fig. 3. It consists of four main usecases: `Fuzzy Parameterizing`,

Table 1. Fifteen selected decision parameters

No	Category	Parameter	Coef. value	Explanation
1	General location	Population	0.07	Number of area population
2		Location	0.10	Distance to hustle center
3		Market statistic	0.08	Number of store surrounding
4	Position of site	Size of site	0.07	Size of site or area
5		Convenience	0.11	Convenience level judged by citizen
6		Visibility	0.11	Visibility level judged by citizen
7	Demographics	Age	0.05	Age range of area citizen
8		Income	0.07	Income average of citizen
9		Future growth and development	0.05	Prediction of population growth
10	Traffic information	Traffic patterns	0.04	Pattern of traffic line
11		Traffic counts	0.04	Traffic density
12	Competition	Competitors of location	0.04	Distance to competitor store
13		Proximity to exhibiting location	0.08	Proximity distance to other fast-food restaurant
14	Cost consideration	Cost of construction	0.03	Cost of site construction
15		Cost of improvement	0.05	Cost of site renovation

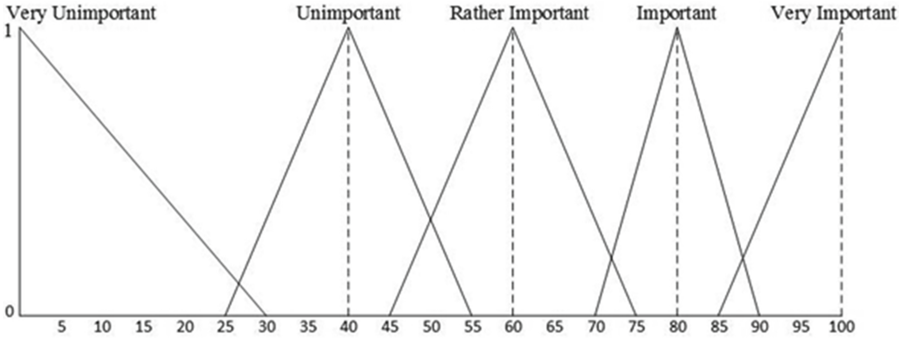


Fig. 2. Fuzzy triangular membership function

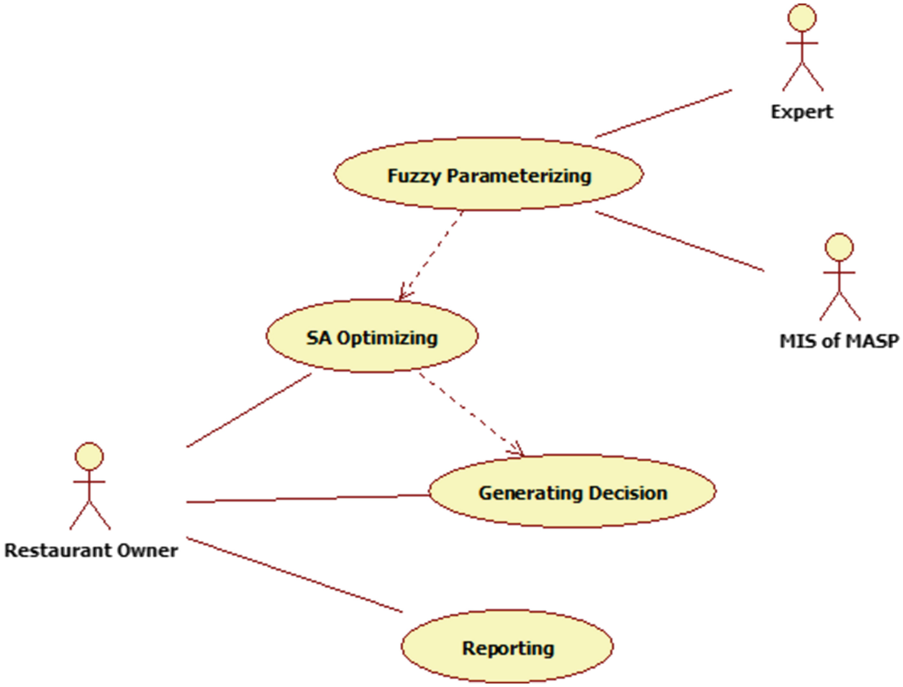


Fig. 3. Usecase diagram of constructed model

`SA optimizing`, `Generating Decision`, and `Reporting`. The usecase `Fuzzy Parameterizing` is interconnected with two types of human and system actor. Human actor `Expert`, as mentioned before, involves in fuzzy-logic process to produce the normal coefficient value; and system actor `MIS of MASP` (Management Information System of Ministry of Agrarian and Spatial Planning of Indonesian Republic) supplies all data reasonably necessitated.

The process of simulated-annealing optimization is technically delivered in usecase SA Optimizing. The human actor Restaurant Owner is using the usecase. Also, the actor Restaurant Owner is able to interact with two other usecases; Generating Decision and Reporting. The owner can get the proposed decision that is possibly used to locate the restaurant site and other information needed from usecases Generating Decision and Reporting respectively.

In generic view, the process of optimizing (based on the method simulated-annealing) can be seen in pseudocode in Code 1. To define the value cPos (current position), a random index of data is done. The index used to extract its parameter value to get cVal (current value) by transferring the parameter to objective function. The variable nVal is taken then, it means a neighbor value; and nVal is compared with bestN, where bestN is the currently best value of a neighbor. For the first time, bestN could be a very small value (for example -999999), as the optimization process will find the biggest one as the most optimal value.

The optimal value seeking is continued by setting cVal as bestN ($cVal \leftarrow bestN$) when $bestN \geq cVal$ occurs. While, when cVal is greater than bestN, the variable delta is operated then. It is used to get the probability from the result of a comparison. The probability is coded by using comparing statement $exp \wedge (-delta/baseTemp) > random(0..1)$. The process will be continued when the condition value is true.

Moreover, the variable baseTemp (in this case is 100) is a control parameter called "adjusted temperature" used to be an indicator to practically control the simulated annealing process. The process will be terminated if the value of the variable baseTemp reaches the value of the variable optTemp, where optTemp is an optimal temperature that has been defined before, the example in this case is 10.

```
Procedure SimulatedAnnealing(procedureParameters) Begin
  <...other variables definition...>
  baseTemp <-- 100
  optTemp <-- 10
  cPos <-- random() //randomizing new parameter combination
  cVal <-- objFunction(cPos)

  While(baseTemp>=optTemp) //looping until local optimum is found
    bestNeighbor <-- getBestNeigh() //finding the best neighbor
    If(bestNeighbor>=cVal) //finding local optimum
      cVal <-- bestNeighbor
      cPos <-- bestNPos
      bestNeighbor <-- 0
    Else
      delta <-- bestNeighbor - cVal
      If(exp^(-delta/baseTemp) > random(0..1)) //Probability
        cVal <-- bestNeighbor
        cPos <-- bestNPos
        bestNeighbor <-- 0
      End if
    End if
    adjust(baseTemp) //adjusting baseTemp
  End while
End
```

Code 1. Pseudocode of Simulated-Annealing Optimization (Utama, 2015)

Furthermore, the high level class diagram of the model is portrayed briefly in Fig. 4. Fundamentally, it consists of five classes; Restaurant Location, Considered Parameter, Decision, SimulatedAnnealing, and MembershipFunction. Fifteen parameters considered are interconnected to class ConsideredParameter. Class Decision is a proposed location as the most objective decision by class SimulatedAnnealing. Class SimulatedAnnealing itself is connecting with class MembershipFunction, where the connection configures that the method used is the combination between methods simulated-annealing and fuzzy-logic.

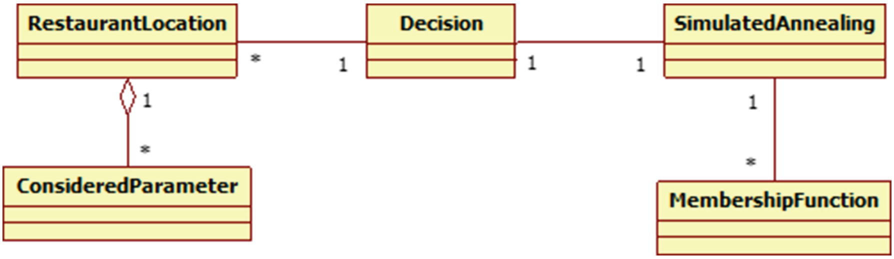


Fig. 4. High level class diagram of the constructed model

Model simulation gave several results. Based on five times of experiment conducted for each type of program looping, it suggested the constant decision alternative that has to be taken. The model also showed bar-chart graph of time for five times of experiment number (see Fig. 5). Program Looping Types 1 – 6 represent 10, 50, 100, 250, 500, and 1000 looping times respectively. Demonstrably, Fig. 6 shows one screenshot of constructed model representing the result of optimization simulation based on 500 looping times.

4.2 Discussion

In using parameters considered in the model, Tzeng et al. (2002) used five aspects and eleven criteria. The aspects included transportation, commercial area, economic, competition and environment. The criteria are about rent cost, transportation cost, convenience to mass transportation system, size of parking space,

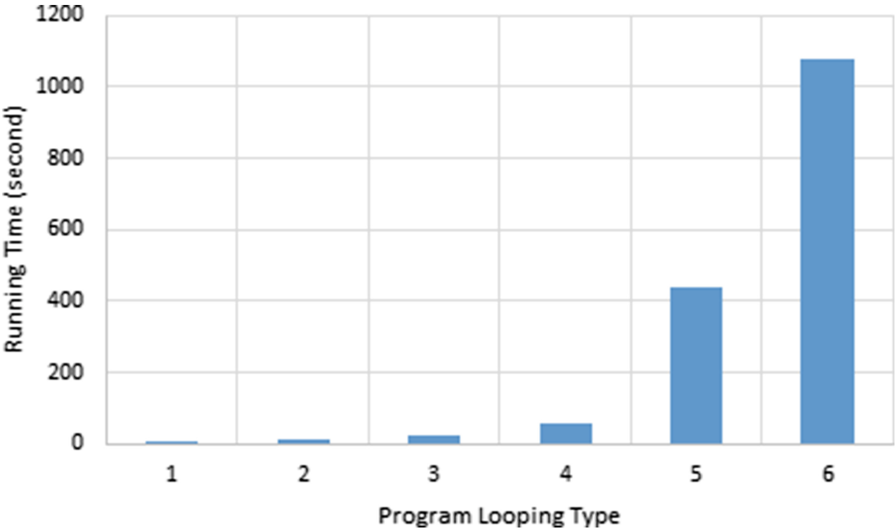
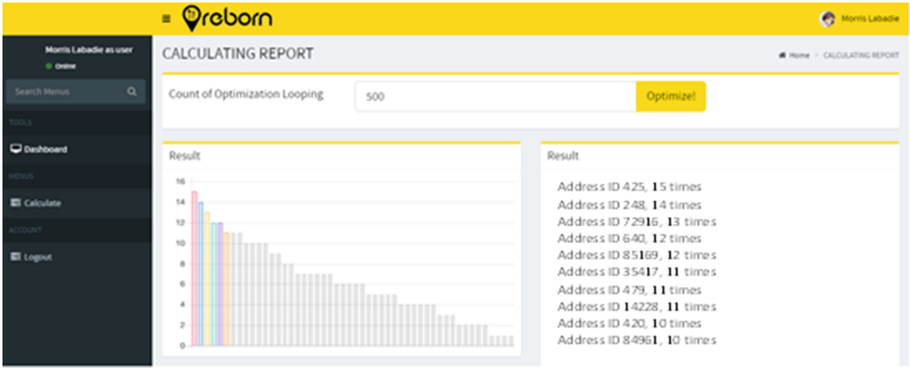


Fig. 5. Program running time for six types of program looping



CALCULATING REPORT

Count of Optimization Looping 500

Result

Result

Address ID 425, 15 times
Address ID 248, 14 times
Address ID 72916, 13 times
Address ID 640, 12 times
Address ID 85169, 12 times
Address ID 35417, 11 times
Address ID 479, 11 times
Address ID 14228, 11 times
Address ID 420, 10 times
Address ID 84961, 10 times

Fig. 6. One sample of screenshots of the constructed model

pedestrian volume, a number of competitors, the intensity of competition, size of the commercial area where the restaurant is located, extent of public facilities, convenience of garbage disposal, and sewage capacity. However, in this study we used six categories with fifteen parameters. Even though, several criteria and parameters said something similar; e.g. pedestrian volume and population, intensity of competition and Competitors of location, etc.

In addition, interestingly, in developing the model, Min (1987) combined behavioral and spatial aspect of location based schemes and systematic sequential decision process. Where, traditionally it only focused on location aspect. Indeed, similarly we officially considered fifteen parameters and mechanically enhanced the use of optimization method to select the best decision alternative.

5 Conclusion and Further Works

Fifteen parameters from six categories have been taken into account in constructing a DSM for placing the fast-food restaurant objectively. Two main methods fuzzy and simulated annealing were combined to correspondingly parameterize the selected parameters and optimize the proposed decision alternative. The graph shows linearly increase of running time when the optimization model was executed until 1,000 program looping.

Several environment parameters could be taken into account for further work, such as Tzeng et al. (2002) offered (convenience of garbage disposal and sewage capacity). They will enrich the model created. Also, combination more than one optimization method is going to build the model valuable and motivating.

References

Černỳ, V.: Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm. J. Optim. Theory Appl. **45**(1), 41–51 (1985)

Chen, L.F., Tsai, C.T.: Data mining framework based on rough set theory to improve location selection decisions: a case study of a restaurant chain. Tourism Manag. **53**, 197–206 (2016)

Dock, J.P., Song, W., Lu, J.: Evaluation of dine-in restaurant location and competitiveness: applications of gravity modelling in Jefferson County, Kentucky. Appl. Geogr. **60**, 204–209 (2015)

Glock, C.H.: Decision support models for managing returnable transport items in supply chain: a systematic literature review. Int. J. Prod. Econ. **183(B)**, 561–569 (2017)

Hong, T., Koo, C., Jeong, K.: A decision support model for reducing electric energy consumption in elementary school facilities. Appl. Energy **95**, 253–266 (2012)

Kim, D., Leigh, J.P.: Are meals at full-service and fast-food restaurants "normal" or "inferior"? Popul. Health Manag. **14**(6), 307–315 (2011)

Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. Science **220**(4598), 671–680 (1983)

Kunsoon, P.: Identification of Site Selection Factors in the U.S. Franchise Restaurant Industry: An Exploratory Study. Master of Science Thesis, Virginia Polytechnic Institute and State University (2002)

Mathiassen, L., Munk-Madsen, A., Nielsen, P.A., Stage, J.: Object-Oriented Analysis and Design. Marko Publishing ApS, Aalborg Denmark (2000)

Min, H.: A multiobjective retail service location model for fastfood restaurants. Omega **15**(5), 429–441 (1987)

Pattanaik, L.N., Yadav, G.: Decision support model for automated railway level crossing system using fuzzy logic control. Procedia Comput. Sci. **48**, 73–76 (2015)

Piltan, M., Sowlati, T.: A multi-criteria decision support model for evaluating the performance of partnerships. Expert Syst. Appl. **45**, 373–384 (2016)

Rosenfeld, Y., Shohet, I.M.: Decision support model for semi-automated selection of renovation alternatives. Autom. Constr. **8**(4), 503–510 (1999)

Sloan, C., Caudill, S.B., Mixon, F.G.: Entrepreneurship and crime: the case of new restaurant location decisions. J. Bus. Ventur. Insights **5**, 19–26 (2016)

Syam, S.S., Bhatnagar, A.: A decision support model for determining the level of product variety with marketing and supply chain considerations. J. Retail. Consum. Serv. **25**, 12–21 (2015)

Thornton, L.E., Lamb, K.E., Ball, K.: Fast food restaurant locations according to socioeconomic disadvantages, urban-regional locality, and schools within Victoria, Australia. SSM Popul. Health **2**, 1–9 (2016)

Tzeng, G.H., Teng, M.H., Chen, J.J., Opricovic, S.: Multicriteria selection for a restaurant location in Taipei. Int. J. Hosp. Manag. **21**(2), 171–187 (2002)

Utama, D.N.: The Optimization of the 3-D Structure of Plants, Using Functional-Structural Plant Models. Case Study of Rice (Oryza sativa L.) in Indonesia. Ph.D Thesis, Georg-August Universität Göttingen (2015)

Zadeh, L.A.: Fuzzy logic = computing with words. IEEE Trans. Fuzzy Syst. **4**(2), 103–111 (1996)

Bayesian Regularization-Based Classification for Proposed Textural and Geometrical Features in Brain MRI

An Approach Towards Automated Tumor Detection

V. Kiran Raj(✉) and Amit Majumder

CBK Infotech India Pvt. Ltd, Bengaluru, India
research.kiranraj@gmail.com

Abstract. Medical image diagnosis plays an important role in many of present days healthcare applications. However, the essence of medical image diagnosis primarily depends on factors such as noise, features and classification accuracy in order to develop efficient methodology towards image segmentation, Image registration and most importantly towards automation of lower level and mid-level image processing. In this paper, a novel system is developed considering Gaussian, speckle and impulse noise models for which an improved adaptive filtering method is proposed for denoising. The novel feature extraction is performed as a two stage process considering the textural and geometrical properties of the image. Finally a classification process using Bayesian regularization back propagation method is used from the procured feature-sets. Experimental results show an improved performance considering image quality assessment metrics and also an improvement in the mean square error is observed as compared to other conventional methods.

Keywords: Gaussian noise · Adaptive filtering · Gabor filter · Bayesian

1 Introduction

A major part which influences the healthcare application is the diagnosis of medical image. The complexity and performance of a system has increased significantly due to technological advancements in Integrated Circuits (IC's), cloud computing and automation. However, these advancements are still uninfluenced with respect to medical imaging. Automation in the medical image diagnosis is inferred to be a critical requirement. i.e. the need for human intervention for diagnosis in medical image is to be compensated. This is especially significant in remote places where, certain decisions could not be taken by the physician due to lack of substantial information. Cootes and Taylor [1] proposed a statistical model for interpreting medical images considering the shape variation and texture variation. Patenuade et al. [2] proposed a Bayesian model for subcortical brain segmentation using attributes related to shape and appearance. Freedman et al. [3] proposed a model based segmentation for medical imagery by using matching distribution. Makni et al. [4] proposed a deformable model combined with a

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_34

probabilistic framework for automatic detection of 3D prostate segmentation based on Magnetic Resonance Imaging (MRI) images. Though there are many geometrical models for effective analysis in medical images, the factor of practicality is still in question. Elakkia and Narendran [5] points out the effects of noise particularly the Gaussian noise in medical imaging.

This is further articulated by Huang et al. [6] by considering the impulse noise along with Gaussian noise model. The filters considered in noise removal were predominantly linear in nature. However, in the case of computed tomography (CT), the influence of Gaussian noise created difficulty in pathological image diagnosis, Hence, an adaptive non-linear approach was considered which led to methods pertaining to Artificial Neural Network (ANN) methods [19] which uses the factor of optimum weight of neural network filter. The factor of optimum weight primarily depends on the feature set extracted from the image. Therefore, this infers the need for methods to perform efficient feature extraction. Effective extraction of features not only enhances the quality of the image but also reduces the energy consumption along with computational and system complexity. Therefore, in this paper a system comprising of denoising, feature extraction and classification is developed and proposed and evaluated.

2 Review of Literature

In this section, a review on the literature considering the issues prevalent in medical image diagnosis and the influence of artificial intelligence in the same is illustrated. The approach considered towards neural network models over other conventional methods used for image denoising, segmentation, registration, etc. proves to be an effective approach both in terms of performance and efficiency. Some of the works are given as follows.

Noreen et al. [7] proposed a segmentation method based on Fuzzy C-means and Discrete Wavelet Transformation (DWT) for MRI based images. The segmented image was further enhanced by Kirch's line/edge detection mask which further improves the quality of the image. When considering the noise which is linear in nature, the inference is that the noise is present in higher frequencies. However, it is observed that the image representation in DWT also lies in the higher frequencies. Hence, differentiating noise from the image content becomes an issue.

Yugander and Sheshagiri [8] proposed a method which uses Multiple Kernel C-Means Clustering (MKFCM) in order to distinguish image into homogenous and non-overlapping closed region. The experimental result shows improved shift invariance and directionality properties, which in turn improved the overall image quality pertaining to visual perception. However, in the analysis based on pattern recognition, the parametric model is more effective in model based approaches; this is more significant in image retrieval and image analysis.

Kumar et al. [9] arrives to such parameters which are obtained using Maximization-Expectation algorithm. The segmentation for the approximate coefficients obtained through DWT is determined from Maximum Likelihood function. The segmentation performed by Maximum Likelihood function achieves a computationally efficient segmentation of image with improves image quality index as compared to other methods such as Finite-Gaussian mixture and Finite-Generalized Gaussian Distribution.

Another major criterion in medical image diagnosis concerning the segmentation process is based on optimization based decision making and information integration dominate which involves analysis of structured morphology. This mainly involves methodologies to find the boundary particularly in deformable models. One such approach was proposed by Staib and Duncan [10]. However, this method addresses the issue of homogeneity, the computational complexity for such methods was still very high.

The issue of homogeneity during image segmentation is emphasized and addressed by Dias and Figueiredo [11] where the case scenarios considered in this context involve supervised which is convex in nature, unsupervised and semi supervised formulations which are non-convex in nature. The method proposed to solve the supervised problem formation was alternating direction method of multipliers; however, in the case of semi supervised and non-supervised problems an Expectation-Maximization algorithm is used. The overall accuracy was 0.66 pertaining to maximum likelihood segmentation.

The complexity of optimization problem was compensated by considering the hidden field approach. This was further enhanced by Figueiredo [12] considering the standard Bayesian approach which is combinatorial in nature (Bayesian Machine learning), the experimental results proves that the proposed method is quite cost effective. The overall inference arrived from the literature review is to address the issue of noise parameter, selection of appropriate feature set and effective classification process to develop an efficient system for image segmentation in medical imaging.

However, the inference made from the above survey is that the necessity to address the pre-processing techniques have been significant and the undermining of the fact that the possibility of achieving a higher classification rate has a dependency towards pre-processing and feature extraction techniques. Hence, in this paper, a novel combination of image denoising as a pre-processing step along with textural and geometrical feature extraction prior to classification process is proposed.

3 Problem Statement

The necessity to address the issue of classification is prevalent, the problem of classification depends factors such as noise estimation and feature-set description. These factors have a direct impact on energy consumption along with system and computational complexity. Therefore, there is a need to develop a segmentation method which considers all these factors. The problem can therefore be stated as "*To develop an image segmentation model considering noise estimation and feature-set selection to improve the performance and effectiveness of classification process from the perspective of automation in imaging diagnostics*". The following section illustrates the proposed system for the same.

4 Proposed System

In this section, the modules involved in developing the system is mainly comprised of three stages which involves

- Developing the noise model and applying denoising method for the same
- Identification and extraction of feature-set on the basis of textural and geometrical data
- Performing classification from extracted features

4.1 Noise Model

The major sources of noise are derived from signals obtained from the sensors. This noise can be further classified as linear and non-linear noise depending on the source (Table 1). The linear noise is defined as the environmental noise to which the humans are susceptible.

Table 1. Types of noise models considered in this experiment

Sl. no	Type of noise	Mathematical representation	Description
1.	Impulse noise	$f(x) = \begin{cases} p/2, for\, x = 0 \\ 1 - p, for\, x = s_{i,j} \\ p/2, for\, x = 255 \end{cases}$	Random irregularities observed between two extremal pixel values generated at constant probability.
2.	Gaussian noise	$PDF_{gauss} = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\sigma-\mu)^2}{2\sigma^2}}$	Combination of true pixel value and the randomized Gaussian distributed noise value.
3.	Speckle noise	$F(g) = \frac{g^{n-1}}{(a-1)!\alpha^{\alpha}} e^{-\frac{g}{\alpha}}$	Coherent interference which is multiplied with original pixel value.

Approach considered towards denoising

The median filter addresses the issue of non-linearity with the use of digital filtering. This is especially prevalent in the case of impulse noise. However, this is not the case in Gaussian and speckle noise. Therefore, estimation of local mean and variance corresponding to each pixel is necessary since, the Gaussian noise is mainly controlled by conditioning the standard deviation. The significance of median filtering is its ability to preserve the edges in an image while maintaining minimum system and computational complexity. The median filter calculates the median which is obtained from the defined pattern from the adjacent pixels in numerical order, consequently the computed median value is replaced with the middle pixel value.

The processed image from the median filter is passed to the adaptive filtering which in this case a wiener filter is considered, as mentioned above the wiener filter is used in the estimation of local mean and variance corresponding to each pixel. The mean computation is represented in Eqs. 1 and 2 as shown below,

$$\mu = \frac{1}{MN}\sum\nolimits_{n_1,n_2\in\eta} a(n_1,n_2) \tag{1}$$

$$\sigma^2 = \frac{1}{MN}\sum\nolimits_{n_1,n_2\in\eta} a(n_1,n_2) - \mu^2 \tag{2}$$

Where, $\eta \rightarrow$ M × N local neighborhood corresponding to each pixel. The filter estimates for the pixel wise wiener filter is given as shown in Eq. 3,

$$b(n_1,n_2) = \mu + \frac{\sigma^2 - v^2}{\sigma^2}(a(n_1,n_2) - \mu) \tag{3}$$

Where, $v^2 \rightarrow$ noise variance.

4.2 Identification and Extraction of Feature-Set

Zainudin et al. [17] illustrates the significance of using Gabor filter as a method for feature extraction in medical images pertaining to tumor detection. This is further upgraded by designing of intensity invariant local image which are used as phase features by improving the Gabor filter [13]. The Gabor transform function is represented as shown in Eq. 4,

$$G_f(\omega,\tau) = \int_{-\infty}^{\infty} f(t)g(t-\tau)e^{-j\omega t}dt \tag{4}$$

For a 2 dimensional Gabor filter Eq. 4 is generalized as shown in Eq. 5,

$$\begin{aligned}\psi_{mn}(xy) = \alpha_0^{-m} exp\left\{\frac{\alpha_0^{-2m}}{8}\left[4\left(xcos\frac{n\pi}{K} + ysin\frac{n\pi}{K}\right)^2 I\omega + \left(-xsin\frac{n\pi}{K} + ysin\frac{n\pi}{K}\right)^2\right]\right\}\\ Xexp[I\omega\omega_0^{-m}\left(xcos\frac{n\pi}{K} + ysin\frac{n\pi}{K}\right)]\end{aligned} \tag{5}$$

Image properties are computed from the obtained feature-set such as centroid, Eccentricity, etc. considered as second order feature-set as shown in Table 2. These second order feature-sets are further identified using Principal Component Analysis (PCA). The identified principle components are then sent as training sets for the classification process which is illustrated in the following section.

4.3 Training and Classification of Feature-Set Using Bayesian Regularization Back Propagation

Using Artificial Neural Network (ANN) for classification of feature-set has been emphasized more due to robustness along with its higher performance efficiency [14, 15]. For the purpose of automated regularization and generalization, a Bayesian

Table 2. Geometrical properties of considered medical MR image

Sl.no	Property	Value
1.	Centroid	[102.97, 115.02]
2.	Eccentricity	0.6100
3.	Euler number	280
4.	Equiv-Diameter	24.3845
5.	Solidity	0.0096

regularization back propagation is considered. Due to its ability of improved mapping function and also possessing an optimum combination of training, learning, and transfer function for classification of feature-set [16] the Bayesian regularization is a preferred choice.

Considering a fixed training set (consisting of 10 neurons in hidden layer and 6 neurons in the output layer) of $\{(x^{(1)}, y^{(1)}), \ldots, (x^{(m)}, y^{(m)}))\}$ for m training sets.

Considering a single training example in the case of batch gradient descent, the objective norm is defined as shown in Eq. 6,

$$J(W, b; x, y) = \frac{1}{2}\left\|h_{W,b}(x) - y\right\|^2 \tag{6}$$

Where, $J(W, b; x, y)$ → average sum of squares error term

$W_{ij}^{(i)}$ → Regularization term which decreases the magnitude of the corresponding weights

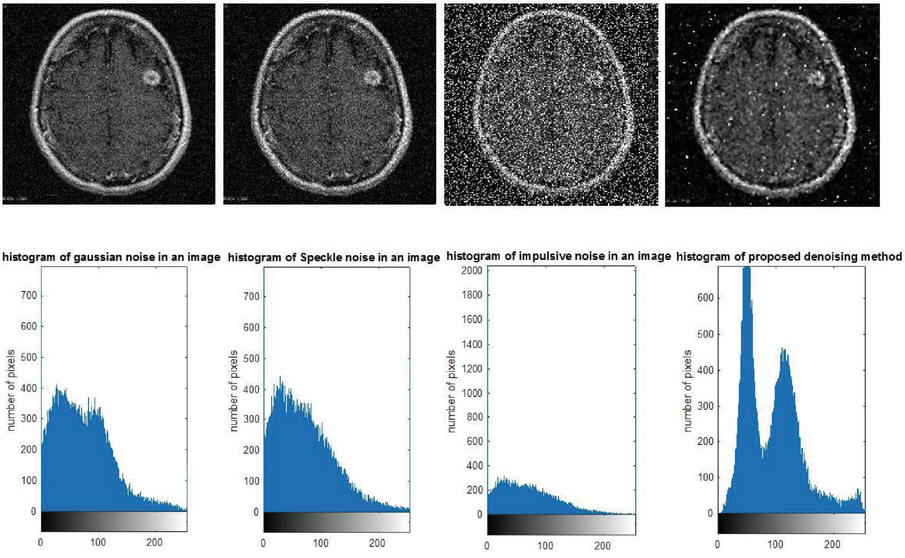
The desired partial derivatives obtained for the defined objective norm $J(W, b)$ is given as shown in Eqs. 7 and 8,

$$\frac{\partial}{\partial W_{ij}^{(l)}} J(W, b; x, y) = a_j^{(l)} \partial_i^{(l+1)} \tag{7}$$

$$\frac{\partial}{\partial b_i^{(l)}} J(W, b; x, y) = \partial_i^{(l+1)} \tag{8}$$

5 Results and Discussions

For an effective classification and feature extraction process, performing denoising for an image is an important pre-requisite. The noise model considered in this experiment are Gaussian, speckle and impulse noise for which a median and an adaptive filter particularly wiener filter is used for denoising process which is shown in Fig. 1. A comparative analysis for evaluation of denoising methods is shown in Figs. 2 and 3 signifies that the proposed denoising method has higher value in terms of PSNR and SSIM compared to other conventional methods.


histogram of gaussian noise in an image
histogram of Speckle noise in an image
histogram of impulsive noise in an image
histogram of proposed denoising method
number of pixels
0
100
200

(a) (b) (c) (d)

Fig. 1. Simulation results concerning noise reduction using proposed denoising method

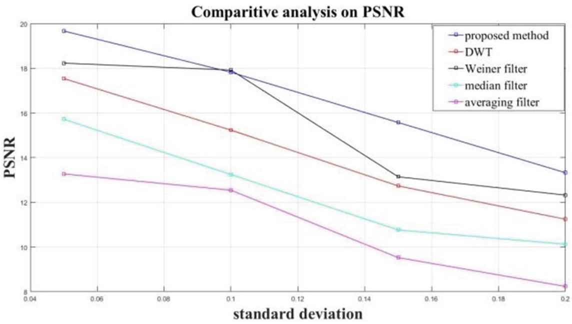


Fig. 2. Comparative analysis for the proposed denoising method on PSNR measure [18]

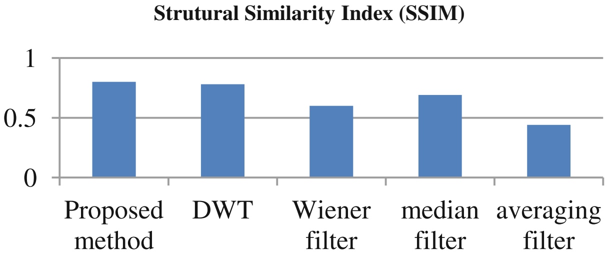


Fig. 3. Comparative analysis for the proposed denoising method on SSIM measure [18]

Peak Signal to Noise Ratio (PSNR)

This measure is considered mainly to indicate the quality of reconstruction for a lossy compression codecs. It is mainly defined by its mean squared error, defined as the ratio of power of reference signal to that of the noisy signal. The mathematical representation for a PSNR is given in Eq. 9,

$$MSE = \frac{1}{mn}\sum\nolimits_{i=1}^{m-1}\sum\nolimits_{j=0}^{n-1}[I(i,j) - K(i,j)]^2 \tag{9}$$

In terms of decibels (dB), The PSNR is defined as shown in Eq. 10,

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right) \tag{10}$$

Structural Similarity Index (SSIM)

This measure is considered for prediction of perceived quality in digital images. In essence, it is used for measuring the similarity between two images. The SSIM is mathematically represented as shown in Eq. 11

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{11}$$

Where,

μ_x, μ_y	→ average of x and y respectively
σ_x^2	→ Variance of x
σ_y^2	→ Variance of y
σ_{xy}	→ Covariance of x and y
c_1, c_2	→ variables to stabilize weak denominator

The Fig. 1 indicates the development of noise model and the consecutive application of denoising techniques. Figure 1(a) indicates the brain MR images considering Gaussian noise which is further developed to possess a speckle noise along with impulsive noise which is shown in Fig. 1(b) and (c) respectively. After applying the proposed denoising method, the resulting de-noised image is shown in Fig. 1(d) in which the lesions (concerning the tumor region) are more visible from the de-noised image.

In this experiment, feature extraction is performed in two stages. The first step involves extracting the textural features using Gabor filter followed by computing image properties (as shown in Table 3). The feature extraction is performed along vertically, horizontally and diagonally along 0^0 and 90^0. The simulation result for the same is shown in Fig. 4.

Table 3. Comparative analysis for the proposed system using Bayesian regularization with other conventional methods

Sl.no	Methods	MSE	Computational time
1.	Proposed method	0.034	35 s
2.	Levinberg Marquardt method	0.025	20 s
3.	Conjugate Gradient Descent	0.032	40 s
4.	Bayesian Regularization	0.015	37 s

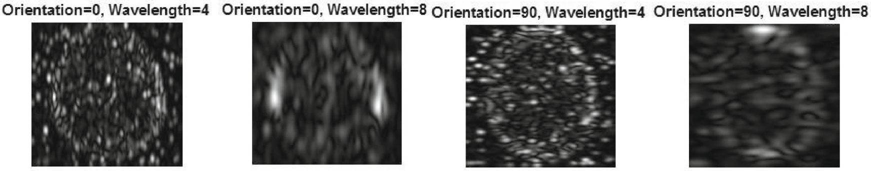

Orientation=0, Wavelength=4
Orientation=0, Wavelength=8
Orientation=90, Wavelength=4
Orientation=90, Wavelength=8

Fig. 4. Feature extraction using Gabor filter considering orientation and wavelength

However, it is observed that after feature extraction at second stages, the data is present in multiple dimensions. Hence, PCA is applied to the feature-set which identifies and extracts the principle components as shown in Fig. 5 below.

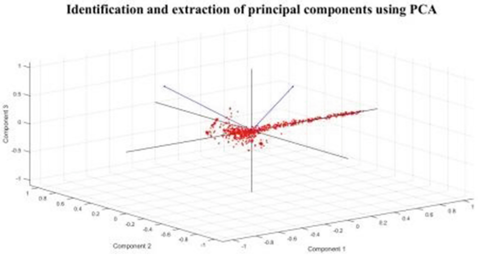


Fig. 5. Principal component identification and extraction of the feature-set

For the purpose of classification analysis, Bayesian regularization back propagation is considered, its performance is measured by the mean square error (MSE) along with computational time (Table 2). The observations concerning MSE and computational time are further indicated as shown in Table 3.

It is observed that the mean square error of proposed method using Bayesian regularization as classification method is considerably less as compared to other conventional methods. However, it is observed that the Levinberg Marquardt method has a considerably lower computational time of 40 s.

6 Conclusion

In this experiment, an improved adaptive filter method is proposed to address the issue of noise particularly in the case of medical image diagnosis along with textural and geometrical image feature extraction. It is observed that the performance of classification using Bayesian regularization back propagation is higher as compared to other conventional methods with respect to MSE and computational time.

References

1. Cootes, T.F., Taylor, C.J.: Statistical models of appearance for computer vision (2004)
2. Patenaude, B., Smith, S.M., Kennedy, D.N., Jenkinson, M.: A Bayesian model of shape and appearance for subcortical brain segmentation. Neuroimage **56**(3), 907–922 (2011)
3. Freedman, D., Radke, R.J., Zhang, T., Jeong, Y., Lovelock, D.M., Chen, G.T.: Model-based segmentation of medical imagery by matching distributions. IEEE Trans. Med. Imaging **24** (3), 281–292 (2005)
4. Makni, N., Puech, P., Lopes, R., Viard, R., Colot, O., Betrouni, N.: Automatic 3D segmentation of prostate in MRI combining a priori knowledge, Markov fields and Bayesian framework. In: 2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 2992–2995. IEEE, 20 August 2008
5. Elakkia, K., Narendran, P.: Survey of medical image segmentation using removal of Gaussian noise in medical image. Int. J. Eng. Sci. **6**, 7593 (2016)
6. Yang, J., Huang, T.: Image super-resolution: historical overview and future challenges. Super-Resolut. Imaging **28**, 1–34 (2010)
7. Noreen, N., Hayat, K., Madani, S.A.: MRI segmentation through wavelets and fuzzy C-means. World Appl. Sci. J. **13**, 34–39 (2011)
8. Yugander, P.: A complex wavelet based image segmentation using MKFCM clustering and Adaptive level set method. In: 2012 International Conference on Advances in Engineering, Science and Management (ICAESM), pp. 297–302. IEEE, 30 March 2012
9. Kumar, H.S., Raja, K., Venugopal, K., Patnaik, L.: Automatic image segmentation using wavelets. Int. J. Comput. Sci. Netw. Secur. **9**(2), 305–313 (2009)
10. Yang, J., Staib, L.H., Duncan, J.S.: Neighbor-constrained segmentation with 3D deformable models. In: Taylor, C., Noble, J.A. (eds.) IPMI 2003. LNCS, vol. 2732, pp. 198–209. Springer, Heidelberg (2003). doi:10.1007/978-3-540-45087-0_17
11. Bioucas-Dias, J.M., Plaza, A., Dobigeon, N., Parente, M., Du, Q., Gader, P., Chanussot, J.: Hyperspectral unmixing overview: geometrical, statistical, and sparse regression-based approaches. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. **5**(2), 354–379 (2012)
12. Seeger, M.W.: Bayesian inference and optimal design for the sparse linear model. J. Mach. Learn. Res. **9**, 759–813 (2008)
13. Li, Z., Mahapatra, D., Tielbeek, J.A., Stoker, J., van Vliet, L.J., Vos, F.M.: Image registration based on autocorrelation of local structure. IEEE Trans. Med. Imaging **35**(1), 63–75 (2016)
14. Boostani, R., Karimzadeh, F., Nami, M.: A comparative review on sleep stage classification methods in patients and healthy individuals. Comput. Methods Programs Biomed. **31**(140), 77–91 (2017)

15. Tan, Y., Zhou, Y., Li, G., Huang, A.: Computational aesthetics of photos quality assessment based on improved artificial neural network combined with an autoencoder technique. Neurocomputing **5**(188), 50–62 (2016)
16. Du, K.L., Swamy, M.N.: Neural Networks and Statistical Learning. Springer Science & Business Media, London (2013)
17. Elnemr, H.A., Zayed, N.M., Fakhreldein, M.A.: Feature extraction techniques: fundamental concepts and survey. In: Handbook of Research on Emerging Perspectives in Intelligent Pattern Recognition, Analysis, and Image Processing, p. 264, 30 November 2015
18. Arakeri, M.P., Reddy, G.R.: A comparative performance evaluation of independent component analysis in medical image denoising. In: 2011 International Conference on Recent Trends in Information Technology (ICRTIT), pp. 770–774. IEEE, 3 June 2011
19. Lin, K.C., Zhang, K.Y., Huang, Y.H., Hung, J.C., Yen, N.: Feature selection based on an improved cat swarm optimization algorithm for big data classification. J. Supercomput. **72** (8), 3210–3221 (2016)

The Comparison of Effects of Relevant-Feature Selection Algorithms on Certain Social-Network Text-Mining Viewpoints

Jan Žižka(✉) and František Dařena

Department of Informatics, FBE, Mendel University in Brno,
Zemědělská 1, 613 00 Brno, Czech Republic
zizka@mendelu.cz

Abstract. This research addresses a well-known problem in the area of text mining: The high computational complexity caused by many irrelevant features (terms, words), which may play an appreciable role of noise from the classification point of view and non-linearly rule the time and memory requirements. Using a set of real-world textual documents represented by sentiment related to three selected and extensively tracked Internet sources freely written in English, a group of available algorithms (*Gain Ratio, Chi Square, Info Gain, Symmetrical Uncertainty, Winnow, One R, Relief F, Principal Components, SVM, LSA*) applied to discovering relevant features was tested with 10,000, 25,000, and 50,000 social-network entries. All the algorithms provided very similar results concerning looking for the relevant features – typically, only the feature significance rank was slightly different. Except for some slower algorithms, the term-preselecting time ranged from seconds to minutes to a couple of hours. However, after using only a relevant fraction of features instead of all of them, the entry length very considerably decreased by several orders of magnitude, particularly for larger data sets having very high dimensionality degree. Despite the extremely strong reduction of the number of words, the classification accuracy remained the same independently on the relevant-feature selection algorithm choice.

Keywords: Relevant features · Feature selection · Text mining · Computational complexity reduction · Classifier training time · Social networks

1 Introduction

From the efficiency point of view, one of typical today's tasks is the automatic analysis of various textual data continuously emerging in the Internet world. The textual data sources are represented by discussion groups, blogs, customers' evaluations of goods or services, electronically published comments of various experts, and so like. This investigation deals with customers' reviews of e-shops and accommodation-booking services, plus economist experts' comments dealing with changes of stock prices. After using a service or monitoring a business

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_35

situation, a customer may write her/his opinion via a mediator, which in this case are *amazon.com*, *booking.com,* and *finance.yahoo.com* [1–3]. The publicly accessible opinions are typically written freely in natural languages and represent a potentially valuable source both for the service providers and text-mining researchers. The specific standpoints, expressed in many relatively short textual documents, can usually be categorized to provide information and – after generalization – valuable knowledge giving a true picture of the activity, which can be later used, for instance, for modifications leading to improvements and raised competitiveness.

Expressing such opinions is very easy for anyone who can use an Internet browser, however, because such data is typically human-like, it is not an easy task for machines (computers) to process it. The more reviews are available, the more valuable knowledge can generally be mined from such a data. Unfortunately, large high-dimensional instances containing too many irrelevant features (sometimes called in this area also as attributes, variables, phrases, n-grams, terms, words) are often strongly weakening the knowledge discovery process because they are effective as noise. A high number of non-relevant features also negatively influences computational complexity (many algorithms have quadratic or higher time complexity due to the big number of dimensions) and memory consumption by data mining algorithms.

Using the Reuters news data, it was shown in [4] that reduction of the number of features thus has became a necessary step in analyzing large amount of data [5]. For example, [6] reported that about 50–90% of features might be removed while maintaining classification accuracy. Usually, a big majority of features has a quite negligible significance for the categorization process. This was also demonstrated in literature for the same hotel-service data-set in [7], where the experiments had to be untimely terminated because of reaching 500 CPU hours for training the professional classifier *c5* [8,9]. Despite the high computational efficiency of *c5*, that very long time (three weeks) was reached already for 160,000 reviews and it was purposeless to continue with even larger data-sets. However, as the authors demonstrated, the generated decision trees used surprisingly only 27 relevant words (for a data-set with 2000 reviews) up to 206 words (for 160,000 reviews).

There exist many feature selection methods and approaches (supervised, unsupervised, semi-supervised) that were variously evaluated with respect to different types of data [11]. The choice of the best method is, however, always dependent on a given specific problem (including specific data for a given domain) [5]. In this paper, a significant possible time-gain reached by eliminating redundant words of typical Internet-based textual document type via removing irrelevant features is described in the following sections.

2 Data and its Preparation

The text-mining problem of investigation how much the use of only preselected relevant data-instance features could affect results was based on analyzing real-world textual data publicly available on the widely popular, freely accessible amazon.com, booking.com, and finance.yahoo.com web-sites. Such a data symbolizes often occurring Internet cases, having many document items, which – after the standard transformation based on word frequencies [12] – were in the investigation represented by typical very sparse vectors. The vectors sparsity come from the fact that the dictionary of each mentioned data source contained tens of thousands terms while a particular textual document had only a tiny fraction of the corresponding vocabulary.

The documents were labeled as positive and negative as their authors had a chance to express their positive and negative opinion related to a given topic. Before starting the analysis, the documents were freed from special characters, numbers, or punctuation (often semantically quite meaningless [12]). Then all the remaining letters were converted to the lowercase form. Such kind of preprocessing is common because it positively (from the computational complexity point of view) decreases the number of lexical units without any significant loss of useful information (for a machine). After that, the words in the reviews were transformed into numbers defined as frequencies of particular words in individual documents. Each document was represented as a vector in a multidimensional vector space defined by all the reviews' words, where each dictionary word constituted an axis (dimension). A vector was a row in a matrix (table), where each column represented a word as a feature. A vector's coordinates were defined by its word frequencies in a given review. Non-zero frequency coordinates were mostly =1, not often 2, and only very seldom 3 or more.

3 Experiments and Their Results

To apply the relevant feature selection process to the given data-set, a collection of implemented tools in the popular WEKA data-mining system [13] was employed (version 3.6). This Java-based software system offers a row of various algorithms, including the feature selection ones. The reason for employing WEKA was also to use the same algorithm-implementation source to obtain comparable results between the selected algorithms.

3.1 Algorithms and Methods

In the first phase, described in this paper, the faster algorithms (*Gain Ratio, Chi Square, Info Gain, Symmetrical Uncertainty*, and *Winnow*) were chosen. Then, much slower algorithms as *One R* and *Relief F* (with several different numbers of nearest neighbors) were tested, too. Finally, some very slow methods (*Principal Components, Latent Semantic Analysis, SVM Attribute Evaluation*) were investigated as well, however, due to their excessively long computational time (for 10,000 documents and more) their results were not ready for publishing.

The details concerning those algorithms can be found in [13] together with references to appropriate mathematical theory, which is not presented here. Each algorithm was applied to the same data-sets for 10,000, 25,000, and 50,000 reviews to study the influence of the data volume (document number) on the result. In addition, the *Relief F* algorithm, which is based on the *k-Nearest Neighbor* method, k-NN, was tested also with different k's (the number of nearest neighbors, that is, the most similar documents); here, because of saving space, only the results for $k = 1$ and $k = 3$ are given but the higher k's (5 and 10) did not provide any better results.

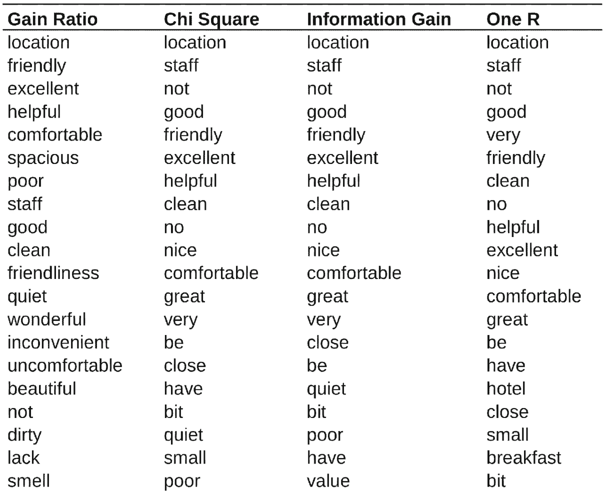
The mentioned algorithms select and rank the individual features according to their contribution to assigning correctly an appropriate label (class name) to an unlabeled item. Many terms can appear in (almost, or completely) all defined classes for a given data set, however, their contribution to the correct labeling depends mainly (with the exception of certain algorithms as. e.g., *Relief F*) on their frequency related to individual classes, which is generally used for computing probabilities transformed into weights. Those weights are finally applied to the ranking of the terms from the most contributing to the least contributing ones. Then, a user can select which terms are significant for her/his application. Omitting the insignificant terms can result also in a strong decrease of computational complexity (time and memory) together with discovering those features that play an important role for a given task. Note that such important features are not discriminated between the given classes – to find which features are important for individual classes is another task in the complex data-mining process.

In this described study, the data sets were generated by random selections from large data collections. The indispensable classes were available for the *amazon* and *booking* data sets: The individual customers' reviews had their categorization usually given by a number of stars, for example, one star meant the worst and five stars the best opinion. For this study, only the worst and best opinions were used, without mixed ones. The *yahoo* messages did not have such categorization available because they existed just as a parallel line with the line of changes of stock prices represented as a time series. Here, two categories were defined: *ascending* and *descending* stock prices. The ascending prices were those ones that pointed up more than 5% with respect to the price moving-average, and vice-versa – more than 5% down defined descending prices. Changes less than 5% were not used here (it could be a third category for another investigation). Note that the goal was not to discover the best classification method.

The experiments run on a standard 64-bit PC machine with 32 GB RAM, SSD, and four double-core processors 3700 MHz.

3.2 Discovered Significant Features

Each tested algorithm was applied to the same data sets and returned a rank of significant words from the most important to the least important ones. The following two figures, Figs. 1 and 2, show only the first 20 features for the *booking* data set having 50,000 documents to save the space because there were thousands of words in each used data set. However, very similar results were obtained for all investigated sets of documents as well as for less document entries (10,000 and 25,000). As a result, a reader can see that different algorithms gave more or less very similar results. Typically, almost the same words were placed at the



Gain Ratio	Chi Square	Information Gain	One R
location	location	location	location
friendly	staff	staff	staff
excellent	not	not	not
helpful	good	good	good
comfortable	friendly	friendly	very
spacious	excellent	excellent	friendly
poor	helpful	helpful	clean
staff	clean	clean	no
good	no	no	helpful
clean	nice	nice	excellent
friendliness	comfortable	comfortable	nice
quiet	great	great	comfortable
wonderful	very	very	great
inconvenient	be	close	be
uncomfortable	close	be	have
beautiful	have	quiet	hotel
not	bit	bit	close
dirty	quiet	poor	small
lack	small	have	breakfast
smell	poor	value	bit

Fig. 1. Booking data: the first 20 top relevant features selected by algorithms Gain Ratio, Chi Square, Information Gain, and One R using 50,000 text documents.

top of the rank, only their order differed. The same was valid for the whole rank but the lower, the bigger order position difference existed – from the probability point of view, the frequency of such words in the given classes was more and more similar, thus providing less information about a word's class-membership support.

Naturally, it is up to a user to select a suitable rank provided by a certain algorithm – however, as the results suggest, the selection is not critical, so that an algorithm that needs the lowest computational complexity may be chosen. The algorithms showed in Fig. 1 were all very fast, taking minutes or less than one hour for larger data sets (25 and 50 thousands of entries).

3.3 Using the Results

For testing the applicability of the relevant feature selection results, this study used a classifier. The goal was to monitor how much the omitting of less significant words might influence the training of a classifier, which often is the following step in the text-mining process.

As a sample classifier, the verified popular Ross Quinlan's *c5* decision tree generator [8] was used. Generally, a decision tree builder's training computational complexity (the upper bound) is given by the formula $O(m, n) = m \cdot n^2$ [14], where m is the number of training samples and n is the number of features. As it can be seen, the complexity rises with the second power of the feature number, therefore decreasing this number can significantly contribute to the complexity decrease. Moreover, filtering out the irrelevant features supports discovering better knowledge hidden in data because such features also play a negative role as noise together with the effect known as the *curse of (high) dimensionality* [10].

Relief F (k=1)	Relief F (k=3)	Symmentrical Uncertainty	Winnow
helpful	helpful	location	not
location	not	staff	location
not	location	friendly	good
clean	clean	helpful	friendly
good	staff	excellent	nice
friendly	good	not	excellent
staff	nice	good	helpful
nice	comfortable	comfortable	clean
comfortable	friendly	nice	great
excellent	excellent	clean	staff
bit	bit	poor	close
spacious	spacious	great	liked
close	small	close	comfortable
really	could	quiet	small
quiet	close	small	more
small	breakfast	citty	noisy
friendliness	friendliness	noisy	poor
could	quiet	be	bit
attentive	great	easy	fantastic
was	attentive	lovely	wonderful

Fig. 2. Booking data: the first 20 top relevant features selected by algorithms Relief F, Symmetrical Uncertainty, and Winnow using 50,000 text documents.

The following figures, Figs. 3, 4, 5, 6 and 7, demonstrate the results for each data set. As a sample, Pearson's χ^2 (Chi Square) applied algorithm outputs were selected, although the other algorithms provided almost the same results. Chi Square was very fast also for the 50,000 data sets, taking some 45 min. After generating the rank of features, only those ones having their importance greater than 0.95 (on the scale 0.0 up to 1.0) were selected for the comparison with results provided using all features.

Consecutively, each triplet of bar charts (sequentially for *amazon*, *booking*, and *yahoo* data set) illustrates the results for 10,000, 25,000, and 50,000 social-network entries. The light-gray bars are for all features (uncut data) and the black ones for the data reduced by Pearson's Chi Square ranking algorithm (the results for all the applied algorithms are not shown here because of the report space limitations, however, they were very similar).

In each triplet, the first chart shows how many unique words were contained in the original dictionaries (light gray bars) and in the reduced dictionaries (black bars) depending on the number of textual entries. It can be seen that especially for 50,000 entries, the reduction was very strong; generally, it was something around $\frac{1}{10}$ for all cases.

The second triplet's chart illustrates the time difference between using the full data (light gray) and the reduced one (black); it was the CPU time of the classifier training process. The difference is very considerable.

The third triplet's chart demonstrates the trained *c5* classification accuracy error for the uncut (light gray) and reduced (black) features – the 10-fold

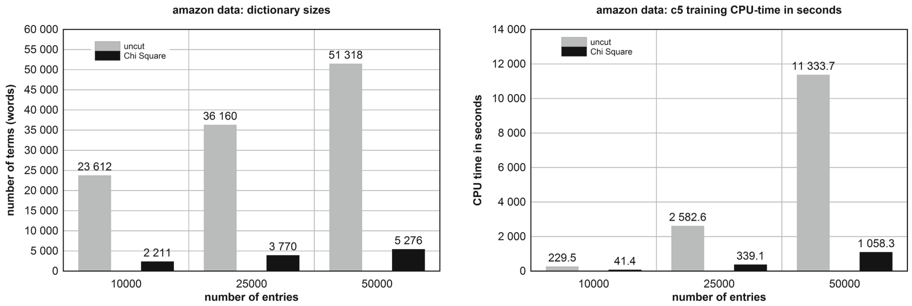


Fig. 3. *Amazon* data: the number of unique terms in data sets (left chart) and the time complexity (right chart).

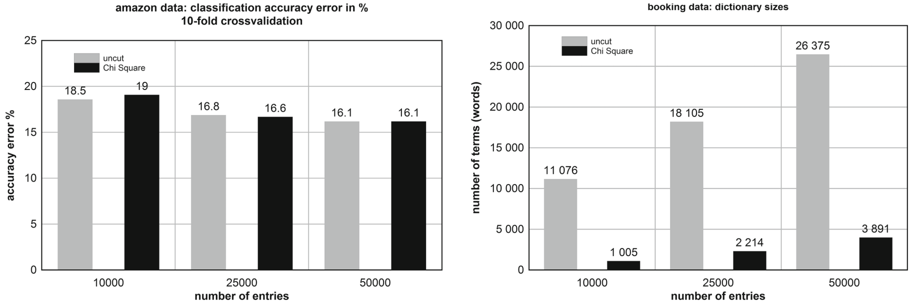


Fig. 4. Left chart: *Amazon* data – the *c5*'s classification accuracy error. Right chart: *Booking* data – the number of unique terms in data sets.

cross-validation measure method provided practically indifferent classification accuracy errors for all three data volumes (number of textual entries). The *c5* decision-tree classifier was here chosen as a typical one. Other classifiers might give different values depending on their principal upper-bound computational complexity $O(.)$.

Note that obtaining the lowest classification error was not a goal here. The relatively high errors for the *yahoo* text documents were given by a rather simple categorization method (mentioned above) – what was important for this study was, however, the fact that filtering out the irrelevant features did not influence significantly the classification results, and, on the other side, it very significantly reduced the training times because of the decreased computational complexity, providing even a small classification improvement (which may, however, be neglected from the practical point of view).

3.4 CPU Time Consumed by the Filtering-Out Process

Returning to the initial problem and looking at the results, one may ask: So, which feature-selection algorithm should be used? Using a little bit more experimental results from the time consumption point of view, all the algorithms

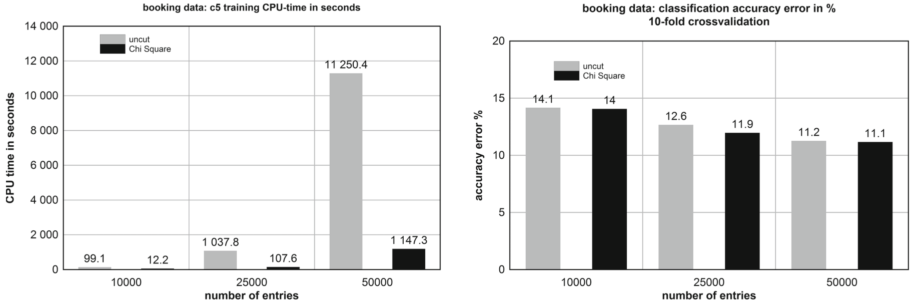


Fig. 5. *Booking* data: the time complexity (left chart) and the *c5*'s classification accuracy error (right chart).

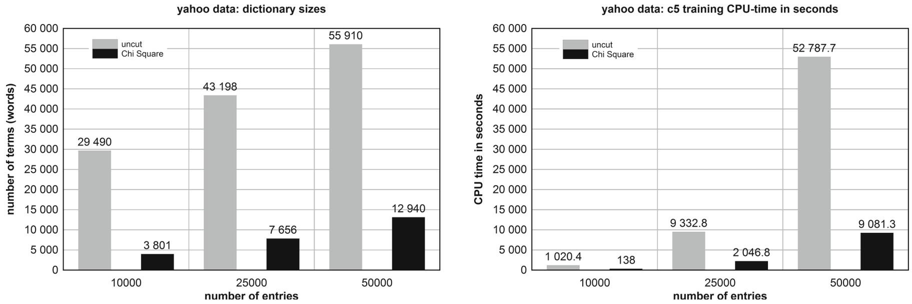


Fig. 6. *Yahoo* data: the number of unique terms in data sets (left chart) and the time complexity (right chart).

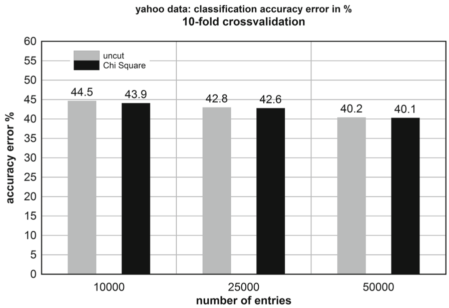


Fig. 7. *Yahoo* data: the *c5*'s classification accuracy error.

provided similar, acceptable results as for the generated relevant words related to the classes. *Gain Ratio* needed less than a minute for 2000 reviews up to 40 min for 50,000. Similarly, *Chi Square* took from 1 to 46 min, *Info Gain* from 1 to 32 min. *One R* and *Symmetrical Uncertainty* were rather slower with 1 to 256 and 1 to 157 min, respectively. The fastest algorithm was *Winnow* with only 1 to 19 min beginning with 2000 and ending with 50,000 reviews, respectively. On the other hand, *Relief F* needed much more CPU time due to the necessary computation of the distance of nearest neighbors,

4 Conclusions

Generating a classifier belongs to the most important text-mining tasks. Revealing the relevant features is a big help but it provides only lists of words that are important for all classes – the discovered knowledge is hidden in combinations of significant words. A branch of a decision tree provides such a combination, and this is why the described research focuses on effective ways to build such kind of knowledge [15].

The experiments showed that it was quite feasible and useful to automatically eliminate irrelevant words before starting the own process of knowledge mining from many textual documents obtained from the Internet. Using a typical textual data composed of many relatively short textual instances created freely in a natural langue (here, English), the results empirically supported the main idea that almost any of available algorithms for separating relevant and irrelevant data features was usable, leading to the almost same results.

As a result, the considerable limitation of computational time devoted to the training of a classifier can be relatively easily reached. Together with certain additional linguistic data preprocessing like excluding stop-words (not used here), it would be possible to expect more efficient text mining procedure within an acceptable time also for today's Big Data problems related namely to long time series, which was tested here using the *finance.yahoo.com* comments running parallel with stock price changes. However, using a batch of data as presented here is not the best approach, and for the time series, new classification and clustering algorithms are being developed [16]. The described research now continues in looking for methods how to combine relevant feature selection with massive on-line analysis.

Not all possible algorithms and methods of selecting relevant features were tested. For example, a classic procedure that employs *Principal Components* was omitted because it worked extremely slowly – it was not ready after many days. Similarly, another method based on *SVM* (support vector machines) feature evaluation could not be applied to the problem, too, from the same reason. However, the *SVM*-based method as well as *Principal Components* and *LSA* (latent semantic analysis) were tested for the smallest data-set (2000 reviews) and despite its extremely long time of computation, it provided almost the same results as other algorithms mentioned in this paper.

Acknowledgments. This research was funded by the Czech Science Foundation, grant No. 16-26353S "Sentiment and its Impact on Stock Markets".

References

1. Amazon.com (2016). https://www.amazon.com
2. Booking.com (2016). https://www.booking.com
3. Yahoo.com (2016). https://finance.yahoo.com
4. Blum, A.L., Langley, P.: Selection of relevant features and examples in machine learning. J. Artif. Intell. **97**(1–2), 245–271 (1997)
5. Dessi, N., Pes, B.: Similarity of feature selection methods: an empirical study across data intensive classification tasks. Expert Syst. Appl. **42**(10), 4632–4642 (2015)
6. Yang, Y., Pederson, J.O.: A comparative study on feature selection in text categorization. In Proceedings of the Fourteenth International Conference on Machine Learning, pp. 412–420 (1997)
7. Žižka, J., Svoboda, A.: Customers' opinion mining from extensive amount of textual reviews in relation to induced knowledge growth. J. Acta Univ. Agric. Silvic. Mendelianae Brun. **63**, 2229–2237 (2015)
8. Data mining tools See5 and C5.0. RuleQuest Research (2016). https://www.rulequest.com/see5-info.html
9. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann, New York (1993)
10. Bellman, R.E.: Dynamic Programming. Counter Dover Publications (2003)
11. Tang, J., Alelyani, S., Liu, H.: Feature selection for classification: a review. In: Aggarwal, C.C. (ed.) Data Classification: Algorithms and Applications, pp. 37–64. CRC Press (2014)
12. Sebastiani, F.: Machine learning in automated text categorization. ACM Comput. Surv. **34**(1), 1–47 (2002)
13. Witten, I.H., Frank, E., Hall, M.A.: Data Mining: Practical Machine Learning Tools and Techniques. Data Transformations. Morgan Kaufmann, San Francisco (2011). Chap. 7
14. Chikalov, I.: Average Time Complexity of Decision Trees. Intelligent Systems Reference Library, vol. 21. Springer, Heidelberg (2011)
15. Dařena, F., Žižka, J.: Interdependence of text mining quality and the input data preprocessing. In: Silhavy, R., Senkerik, R., Oplatkova, Z.K., Prokopova, Z., Silhavy, P. (eds.) Artificial Intelligence Perspectives and Applications. AISC, vol. 347, pp. 141–150. Springer, Cham (2015). doi:10.1007/978-3-319-18476-0_15
16. Bifet, A., Holmes, G., Kirkby, R., Pfahringer, B.: MOA: massive online analysis. J. Mach. Learn. Res. **11**, 1601–1604 (2010)

An Improved Speaker Identification System Using Automatic Split-Merge Incremental Learning (A-SMILE) of Gaussian Mixture Models

Ayoub Bouziane(✉), Jamal Kharroubi, and Arsalane Zarghili

Laboratory of Intelligent Systems and Applications,
Faculty of Sciences and Technologies, Fez, Morocco
ayoub.bouziane@usmba.ac.ma,
jamal.kharroubi@gmail.com, zarghili@yahoo.fr

Abstract. In this paper, a new model-based clustering algorithm is introduced for optimal speaker modeling in speaker identification systems. The introduced algorithm can estimates the optimal number of mixture components using a cross-validation methodology, as well as, overcome the initialization sensitivity and local maxima problems of classical EM algorithm using a split & merge incremental learning approach. The performed experiments in speaker identification task demonstrate the efficiency and effectivity of the proposed algorithm compared to the commonly used Expectation-Maximization (EM) algorithm.

Keywords: Speaker identification · Speaker modeling · Gaussian Mixture Models (GMMs) · Expectation maximization (EM) algorithm · Split and merge EM algorithms · Incremental learning of GMMs · Auto-clustering

1 Introduction

The Gaussian Mixture Models were firstly introduced to the speaker recognition community by Reynolds in 1995 [1]. Since then, they have become the predominant approach for speaker modeling in text-independent speaker recognition systems, and the basis of the most successful approaches that have been emerged in the last decade. The GMM-UBM based approach [2, 3], the hybrid GMM-SVM approach [4], the joint factor analysis approach [5], and the recently introduced i-vector approach [6] are all based on Gaussian mixture models. The motivation behind the use of Gaussian mixture models for speaker modeling is generally based on the assumption that Gaussian densities may model a set of hidden acoustical classes that reflect some general speaker dependent vocal tract characteristics.

The parameter estimates of the GMM are generally obtained using the maximum likelihood (ML) estimation method via the Expectation-maximization (EM) algorithm. However, despite of its conceptual simplicity, the standard EM algorithm suffers from two main drawbacks: (1) it depends heavily on the choice of initial model parameters which may lead the algorithm to a local solution and not necessarily the global one,

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_36

(2) it requires to predefine the number of mixture components manually, which is not often possible in practice.

In order to overcome the problems of initialization sensitivity and local optimum convergence, Ueda and Nakano [7] were proposed the deterministic annealing EM algorithm by reformulating the problem of maximizing the log-likelihood - in the classical EM algorithm - as a problem of minimizing the thermodynamic free energy defined using the maximum entropy principle and statistical mechanics analogy.

Later on, Ueda and Nakano [8] incorporated the split-and-merge operations into the EM algorithm (SMEM). The mixture model is trained firstly using the classical EM algorithm and optimized afterwards using a sequence of split and merge operations. Zhang et al. [9], proposed a modified Split-and-Merge EM algorithm (MSMEM) for speaker verification tasks, where the used criteria for selecting split-and-merge candidates are adapted to the speaker recognition context.

Several incremental learning methods have been also proposed in this context. Those methods start from a single component and iteratively adding new components to the mixture model, until the desired number of components is reached, either via splitting procedure [10] or using some heuristic search method [11]. However, the above proposed methods only optimize the model parameter estimates with a given number of components.

In an attempt to learn automatically the number of components, various traditional approaches using some cost-function based criteria have been proposed for selecting the appropriate number of components (e.g. Akaike's information criterion (AIC) [12], Minimum Description Length (MDL) [13], integrated classification likelihood (ICL) criterion [14], Bozdogan's index of informational complexity (ICOMP) [15]). In [16], a Variation Bayesian analysis (VBA) - based method was proposed to estimate the optimal number of components in the UBM mixture model. A cross-validation strategy using the so-called Cross-validated likelihood [17] has been proposed for automatically determining the appropriate number of components in finite mixture modeling. Lee et al. [18] were proposed an incremental learning method based on the mutual relationship between Gaussian components.

In this paper, we propose a new auto-clustering algorithm for Gaussian Mixture Models training. The main idea of this algorithm is based on the successful incremental split and merge learning methodology of the SMILE Algorithm [10] and a cross-validation methodology for the selection of the optimal number of Gaussian components. For evaluation simplicity, the proposed algorithm is assessed in speaker identification task using a simple GMM based SI system.

The remainder of this paper is organized as follows. Firstly, the general operating structure of speaker identification systems is briefly highlighted. Secondly, a brief description of the speaker modeling process using Gaussian Mixture Models is given. Next, the principle of split-merge incremental learning of Gaussian Mixture Models is explained. Afterwards, the proposed model-based clustering algorithm is introduced. Thereafter, experimental results and discussions are presented in the sixth section. Finally, conclusions, implications of the results and future research directions are provided in the last section.

2 The General Operating Structure of the Speaker Identification Systems

Contemporary speaker identification systems are generally composed of two principal components: The feature extraction component and the speaker modeling component [19]. The feature extraction component involves the processing of speech signal and the extraction of speaker's characteristics, while the modeling component aims to train the speakers' models on the basis of extracted characteristics.

The basic operating structure of speaker identification systems, as shown in Fig. 1, is composed of two distinct phases: the training phase and the testing phase. During the first phase, i.e. training or enrollment phase, speech samples are collected from all speakers, their respective features vectors that summarize the characteristics of their vocal tracts are extracted and they are used to build or train a reference model for each speaker. While in the second phase, i.e. testing phase, the speech signal of the unknown speaker is acquired, corresponding feature vectors are extracted. The extracted feature vectors are then compared, within the pattern matching module, with the known speaker's models already trained and stored - during the training phase - in the system database. The similarity or matching scores computed on the basis of this comparison are finally used to make a decision about the identity of the unknown speaker.

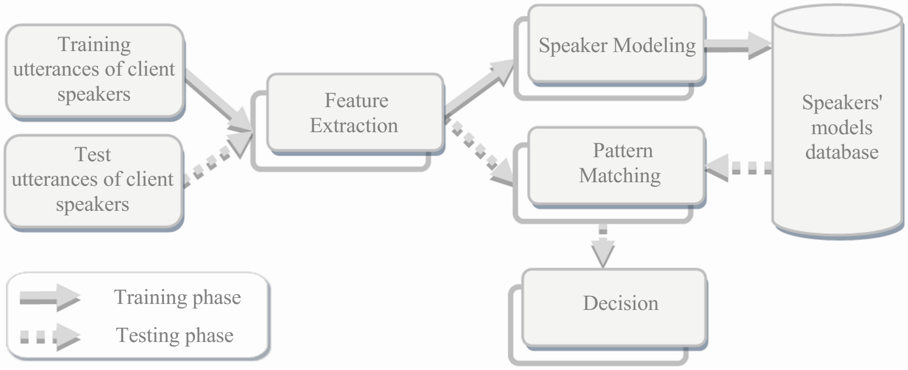

Training utterances of client speakers
Test utterances of client speakers
Feature Extraction
Speaker Modeling
Pattern Matching
Speakers' models database
Decision
Training phase
Testing phase

Fig. 1. The basic framework and components of speaker recognition systems

3 Speaker Modeling Using Gaussian Mixture Models

The basic idea underlying the GMM approach lies in modeling the distribution of speaker's feature vectors by a Gaussian mixture density. The Gaussian mixture density is generally defined by a weighted sum of M component Gaussian densities, as given by the equation:

$$P(x_t|\lambda) = \sum_{i=1}^{M} w_i b_i(x_t) = \sum_{i=1}^{M} w_i g(x_t|\mu_i, \Sigma_i) \quad (1)$$

where x_t is a D-dimensional speech feature vector, w_i, $i = 1, 2, 3, \ldots, M$ are the mixture weights and $b_i(x) = g(x|\mu_i, \Sigma_i), i = 1, 2, 3, \ldots, M$ are the Gaussian densities. Each component density is a D-variate Gaussian function of the following form:

$$g(x_t|\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} exp\left\{ -\frac{1}{2}(x_t - \mu_i)^T \Sigma_i^{-1}(x_t - \mu_i) \right\} \tag{2}$$

The complete Gaussian mixture density λ of each speaker is parameterized by the collection of the mean vectors, covariance matrices and mixture weights of all component densities $\lambda = \{w_i, \mu_i, \Sigma_i\}, i = 1, 2, \ldots, M$. The mixture weights, w_i, furthermore satisfy the constraint $\sum_{i=1}^{M} w_i = 1$.

The model parameters $\lambda = \{w_i, \mu_i, \Sigma_i\}, i = 1, 2, \ldots, M$ are determined in such a way that they best match the distribution of the training feature vectors $X = \{x_1, x_2, \ldots, x_T\}$ i.e. the model parameters $\lambda = \{w_I, \mu_I, \Sigma_i\}$ must maximize the log-likelihood of the GMM, given the training data X: log $P(X|\lambda) = \sum_{t=1}^{T} \log P(x_t|\lambda)$.

The classical method used in this context is the maximum likelihood estimation (MLE) method via the Expectation-Maximization (EM) algorithm. The basic idea of the EM algorithm, as shown in Algorithm 1, is to start with an initial model λ and tends to estimate a new model $\bar{\lambda}$, such that: $p(X|\lambda) \geq p(X|\bar{\lambda})$. $\bar{\lambda}$ then becomes the initial model for the next iteration and the process is repeated until an increase in the log-likelihood of the data, given the current model, is less than some convergence threshold.

Algorithm 1. The EM algorithm

```
Step 0:Initialize the model parameters.
Repeat:
Expectation Step: Compute the a posteriori probability$P(i|x_t, \lambda)$:
    $P(i|x_t, \lambda) = \frac{w_i g(x_t|\mu_i, \Sigma^2{}_i)}{\sum_{k=1}^{M} w_k g(x_t|\mu_k, \Sigma^2{}_k)}$    (3)
Maximization Step: Re-estimate the new model parameters, i.e. the mixture weights, the means and variances vectors, using the following a:
    $\widehat{w}_i = \frac{1}{T}\sum_{t=1}^{T} P(i|x_t, \lambda); \quad \hat{\mu}_i = \frac{\sum_{t=1}^{T} P(i|x_t, \lambda) x_t}{\sum_{t=1}^{T} P(i|x_t, \lambda)}; \quad \widehat{\Sigma^2}_i = \frac{\sum_{t=1}^{T} P(i|x_t, \lambda) {x_t}^2}{\sum_{t=1}^{T} P(i|x_t, \lambda)} - \hat{\mu}_i{}^2$    (4)
Until: Convergence.
```

In speaker identification, the objective of the system is to identify the person speaking in a speech utterance X_U, given a group of speakers $S = \{1, 2, \ldots, M\}$ represented by Gaussian Mixture Models $\{\lambda_1, \lambda_2, \ldots, \lambda_M\}$. The feature representation of the test utterance $X_U = \{x_1, x_2, \ldots, x_T\}$ is compared against the set of speaker models $\{\lambda_1, \lambda_2, \ldots, \lambda_M\}$, and the speaker whose model best matches X is returned:

$$\hat{S} = arg \max_{1 \leq s \leq M} log P(X_U|\lambda_s)$$

where $log\, P(X_{Test}|\lambda_s)$ is the log-likelihood that the utterance X_{Test} belongs to the trained speaker s.

4 Split & Merge Incremental Learning (SMILE) of Gaussian Mixture Models

The main idea of the Split & Merge Incremental Learning methodology, as proposed by Blekas [10], is to start with a mixture model of two Gaussians components and iteratively splits and merges the Gaussian components until a specified number of components is reached. During each iteration, a Gaussian component is selected and splitted into two components. Next, the mixture model parameters are optimized using partial and full EM algorithms, respectively [8]. Afterward, two Gaussian components of the optimized model are selected and merged into one component. Thereafter, the resulted model will be the object of an optimization operation using the partial and the full EM, respectively. If this merge operation leads to a mixture model of likelihood greater than that of the initial model, we accept the resulted mixture model. Otherwise, we choose the model created just after the Split & Optimization operations, which corresponds to a mixture model with an additional component. This procedure is then repeated until the specified number of components is reached. The Split & Merge Incremental Learning (SMILE) algorithm can be summarized as follows:

Algorithm 2. The Split-Merge Incremental Learning (SMILE) algorithm

```
Input:The data set X and the number of Gaussian components K.
Output: Gaussian mixture Model parameters.
Step 0: Start with $K=2$.
Repeat:
  1. Estimate the current log-likelihood $L_1$
  2. Perform SOMO ( Split&Optimization& Merge&Optimization) operation
     ▪ Split operation: Select a Gaussian component $J^*$ and divide it into two com-
     ponents $J_1^*$ and $J_2^*$.
     ▪ Optimization operation: Perform partial-EM and then full EM.
     ▪ Merge operation: Select two Gaussian components $k_1$ and $k_2$ and merge
     them.
     ▪ Optimization operation: Perform partial-EM and then full EM.
  3. Estimate the log-likelihood $L_1$ of the new model and compare it with$L_2$.
     ▪ If$L_2 > L_1$: Accept the fused Gaussian component, set $L_1 = L_2$and go to
     step 2.
     ▪ Else: Reject the last merge operation and set $k = k + 1$.
Until:$k = K$
```

The candidate components for splitting or merging can be selected using several criteria. Blekas et al. [10], have suggested three criteria for the selection of the components to be splitted (Entropy-based criterion, Local Log-likelihood based criterion and local Kullback divergence based criterion), and two criteria for the selection of components to be merged (Symmetric Kullback Leibler based criterion and the distribution overlap based criterion).

5 Automatic Split & Merge Incremental Learning (A-SMILE) of Gaussian Mixture Models

Although the SMILE algorithm has been shown to deal successfully with the initialization sensitivity and local maxima problems of the commonly used Expectation-Maximization (EM) algorithm, it is still limited by its necessity to a predefined number of components, which is often unknown in practice. In this section, we describe our proposed algorithm to estimate the optimal number of Gaussian components, while keeping the Incremental Split and Merge Learning methodology of the SMILE Algorithm.

The main idea of the proposed algorithm is based on the principle of the split-merge incremental learning of Gaussian mixture models for optimizing the model parameters estimates and a cross-validation methodology for the selection of the optimal number of Gaussian components.

The Cross-Validation (CV) is one of the popular techniques used for model selection and adjustment. It is used for selecting the optimal model which fits the training data more accurately, without under-fitting neither over-fitting. The main idea behind the Cross-validation technique is to divide the training data into two sets, one of which is used to build the model and the remaining set is used to evaluate or validate the built model using an appropriate score function. In fact, various cross-validation methodologies have been distinguished in the literature, including the holdout method, K-fold cross-validation and the Leave-one-out cross-validation methods. The most commonly used methodology is the k-fold Cross-validation.

In k-fold Cross-validation, the training data $X = \{x_1, x_2, \ldots, x_T\}$ is randomly partitioned into K roughly equal-sized sets, *K−1* sets are used to build the model and the remaining set is used to evaluate or validate the built model. The cross-validation process is then repeated k times, and every data point must to be in a test set exactly once and gets to be in a training set k−1 times. Finally, the model that optimizes the most the scoring function is selected as the best model.

In the context of probabilistic model-based clustering, Smyth [17] has proposed the so-called cross-validated likelihood as a scoring function for automatically determining the appropriate number of components (given the data) in a mixture model. The cross-validated likelihood can be defined as:

$$CvL^M = \frac{1}{K}\sum_{i=1}^{K} L_i^M = \frac{1}{K}\sum_{i=1}^{K} P\left(\lambda_{X \setminus S_i}^{(M)} | S_i\right) \tag{5}$$

where $\lambda_{X \setminus S_i}^{(M)}$ denotes the M-order mixture model whose parameters were estimated from the K−1 training sets $\{\bigcup_{j \neq i} S_j / 1 \leq j \leq K\}$. The M-order mixture model which maximizes the cross-validated likelihood CvL^M is chosen as the best model.

Algorithm 3. The proposed automatic SMILE algorithm

```
Input: The dataset $X$.
Output: Gaussian mixture Model parameters with the optimal number of components.
❖ Step 0:
      • Start with $M=2$.
      • Randomly divide the dataset $X = \{x_1, x_2, ..., x_T\}$into K roughly equal sized
        sets $S_{1\le i\le K}$.
      • For each $1 \le i \le K$, estimate the model parameters $\lambda^{(2)}_{X\backslash S_i}$and calculate the
        test likelihood $L_i^2$d.
      • Estimate the Cross-validated likelihood $CvL^2$.
❖ Repeat:
  ▪ For each1 $\le i \le K$do
      • Perform SOMO operations on the mixture model parameters $\lambda^{(M)}_{X\backslash S_i}$until the
        number of components of the resulted Model is increased by one.
      • Calculate the test likelihood $L_i^M$.
  ▪ Estimate the Cross-validated likelihood $CvL^M$.
  ▪ Set $M = M + 1$.
❖ Until: No further increase in the Cross-validatedlikelihood $\mathrm{CvL}^{\mathrm{M}}$.
❖ Select the best Model $\lambda^{(M-1)}_{X\backslash S_b}$that maximizes the most the Cross-validated like-
  lihood $L_i^M$.
❖ Refine the parameters of the selected model using the EM algorithm and the
  overall dataset $X = \{x_1, x_2, ..., x_T\}$ as training data.
❖ Return the refined model parameters.
```

Let $X = \{x_1, x_2, \ldots, x_T\}$ the dataset used for building the Gaussian mixture model. The first step of the proposed algorithm, as shown in Algorithm 1, is to divide the dataset X into K roughly equal sized sets $S_{1 \le i \le K}$. Next, K Gaussian mixture models of two components $\{\lambda^{(2)}_{X\backslash S_i}; 1 \le i \le K\}$ are fitted and trained respectively using the training sets:$\left\{\left\{\bigcup_{j \neq i} S_j; 1 \le j \le K\right\}; 1 \le i \le K\right\}$.

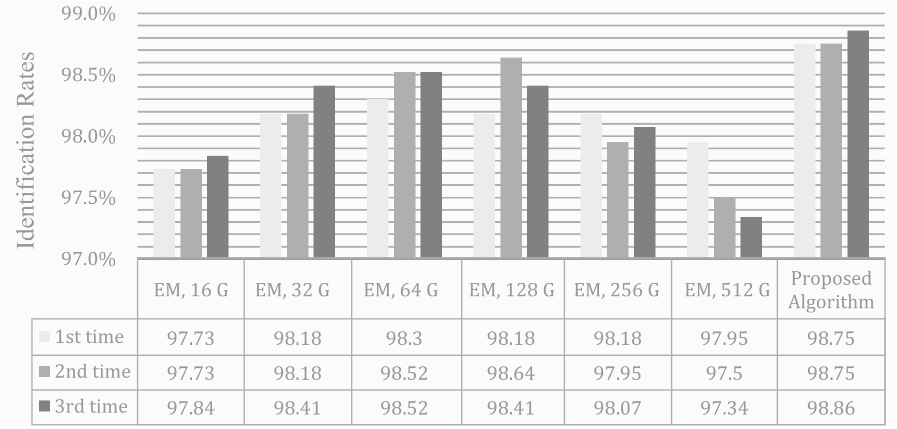
The test likelihood $\mathrm{L}_{\mathrm{k}}^{2}$ of each model $\lambda^{(2)}_{X\backslash S_i}$ is computed and the overall the Cross-validated likelihood CvL^2 is estimated. Thereafter, a sequence of Split & Optimization, Merge & Optimization (SOMO) operations is performed on each model until its number of components is increased by one. The test likelihoods L_k^{2+1} and the cross-validated likelihood CvL^{2+1} are estimated again. If the cross-validated likelihood of the new models $\lambda^{(2+1)}_{X\backslash S_i}; 1 \le i \le K\}$ is superior to that of the old models $\lambda^{(2+1)}_{X\backslash S_i}; 1 \le i \le K\}$, we accept the new models and repeat the process of SOMO operations and cross-validated likelihood evaluation. Otherwise, the new models are rejected and the old model that maximizes the most the cross-validated likelihood is selected. The parameters of the selected model are refined using the classical EM algorithm and the overall dataset $X = \{x_1, x_2, \ldots, x_T\}$ as training data. The refined model is returned as the optimal model.

6 Experiments and Results

The aim of the performed experiments in this section is to assess and evaluate the performance of the proposed automatic clustering algorithm for the text-independent speaker identification task.

The speech corpus used in this study is composed of audio recordings from 40 Moroccan speakers in the age between 18 and 30 years, 17 female and 23 male. Each speaker is recorded for at least more than two sessions separated by approximately two-three weeks. The type of speech includes free monologue in Moroccan dialect and read text in Arabic, French and English languages. The audio recordings were collected from speakers over internet as voice messages through Skype, which is considered as the largest voice over internet provider on the planet. In order to cover a wide range of real-life acoustical environments, we recommended the speakers to make calls from many different places, e.g., home, office etc. Furthermore, various types of recording equipment were used for recording (laptops, tablets, and smartphones …). The voice messages are digitized at 16 kHz with a resolution of 16 bits (mono, PCM) and stored in wav format, which is considered as the most commonly used file type. During the training phase, approximately of 54 s (4 utterances) of speech per speaker were used for building speakers models. While in the testing phase, the evaluation data consisted of five identification sessions (i.e., five tests per speaker each of approximately 8 s in duration). The feature vectors were extracted as Mel-Frequency Cepstral Coefficients [19].

The obtained identification rates using the proposed automatic clustering algorithm and the commonly used EM algorithm with Gaussian mixture models of several sizes are reported in Fig. 2. In order to evaluate the sensitivity of accessed algorithms to initial parameters, the experiments were repeated three times using the same parameters and experimental protocol.



	EM, 16 G	EM, 32 G	EM, 64 G	EM, 128 G	EM, 256 G	EM, 512 G	Proposed Algorithm
■ 1st time	97.73	98.18	98.3	98.18	98.18	97.95	98.75
■ 2nd time	97.73	98.18	98.52	98.64	97.95	97.5	98.75
■ 3rd time	97.84	98.41	98.52	98.41	98.07	97.34	98.86

Fig. 2. The obtained identification rates using the classical EM algorithm with different number of Gaussian components and identification rates obtained using the proposed automatic clustering algorithm.

First and foremost, as it can be seen from the obtained results that the highest identification rates using the EM Algorithm were obtained using model sizes ranging from 32 to 256 Gaussian components. Additionally, it appears that the EM algorithm is very sensitive to initialization of model parameters, and this sensitivity affect the performance of the system. For example, when we have used 128 Gaussian components at the first time we have obtained an identification rate of 98.18%, while when we have repeated the same experiment with the same experimental protocol, the accuracy of the system was increased up to 98.64%.

On the other hand, the obtained results clearly show the significant gain in performance provided by the proposed automatic clustering algorithm compared to the commonly used EM algorithm. Furthermore, it can be seen that our proposed algorithm is less sensitive to initial parameters compared the EM algorithm which depends heavily on the initial parameters. For instance, when we have repeated the experiment for the second time we have obtained the same identification rate, and when we have repeated the experiment for the third time, we notice that the accuracy of the system has not changed much compared to the traditional system.

Moreover, it seems from the successful application of our proposed auto-clustering algorithm - which models each speaker using the appropriate number of Gaussian components- that the number of acoustic classes can differ from a speaker to another. Effectively, an optimal speaker modelization using the appropriate number of acoustic classes for each speaker can improve the performance of speaker recognition systems better than the traditional methods which model the speakers based on the assumption that they have the same number of acoustic classes.

7 Conclusions and Future Research Directions

In this paper, a new model-based clustering algorithm was proposed for optimal speaker modeling in speaker identification systems. The proposed algorithm can automatically estimates the optimal number of Gaussian components using a cross-validation methodology, as well as, optimizes the model parameters estimates using the successful split-merge incremental learning methodology. The obtained results demonstrate the efficiency and effectivity of the proposed algorithm compared to the traditional and commonly used EM algorithm. Additionally, the positively obtained results using the proposed algorithm revealed that speaker acoustics classes can differ from a speaker to another. Indeed, speaker modeling using the appropriate number of acoustic classes can improve the performance of speaker recognition systems.

To further our research, we are planning to validate these findings in a larger speech corpus. Furthermore, we intend to apply our proposed algorithm to optimize the universal background model in the GMM-UBM [2], the Hybrid GMM-SVM [4] and the i-vectors [6] approaches. Moreover, we will concentrate on the optimization of speakers' models within these approaches based on the revealed fact that speakers don't share the same number of acoustic classes.

References

1. Reynolds, D.A.: Automatic speaker recognition using gaussian mixture speaker models. Lincoln Lab. J. **8**(2), 173–192 (1995)
2. Reynolds, D.A., Quatieri, T.F., Dunn, R.B.: Speaker verification using adapted gaussian mixture models. Digit. Sig. Process. **10**(1), 19–41 (2000)
3. Reynolds, D.: Universal background models. In: Li, S.Z., Jain, A.K. (eds.) Encyclopedia of Biometrics, pp. 1547–1550. Springer, New York (2015)
4. Campbell, W.M., Sturim, D.E., Reynolds, D.A.: Support vector machines using GMM supervectors for speaker verification. IEEE Sig. Process. Lett. **13**(5), 308–311 (2006)
5. Kenny, P.: Joint factor analysis of speaker and session variability: theory and algorithms. CRIM, Montreal, (Report) CRIM-06/08-13 (2005)
6. Kenny, P.: A small footprint i-vector extractor. In: Odyssey, pp. 1–6 (2012)
7. Ueda, N., Nakano, R.: Deterministic annealing EM algorithm. Neural Netw. **11**(2), 271–282 (1998)
8. Ueda, N., Nakano, R., Ghahramani, Z., Hinton, G.E.: SMEM algorithm for mixture models. Neural Comput. **12**(9), 2109–2128 (2000)
9. Zhang, Y., Scordilis, M.S.: Optimization of GMM training for speaker verification. In: ODYSSEY 2004 The Speaker and Language Recognition Workshop (2004)
10. Blekas, K., Lagaris, Isaac E.: Split–Merge Incremental LEarning (SMILE) of mixture models. In: Sá, J.M., Alexandre, Luís A., Duch, W., Mandic, D. (eds.) ICANN 2007. LNCS, vol. 4669, pp. 291–300. Springer, Heidelberg (2007). doi:10.1007/978-3-540-74695-9_30
11. Verbeek, J.J., Vlassis, N., Kröse, B.: Efficient greedy learning of gaussian mixture models. Neural Comput. **15**(2), 469–485 (2003)
12. Akaike, H.: Information theory and an extension of the maximum likelihood principle. In: Parzen, E., Tanabe, K., Kitagawa, G., (eds.) Selected Papers of Hirotugu Akaike, pp. 199–213. Springer New York (1998)
13. Schwarz, G., et al.: Estimating the dimension of a model. Ann. Stat. **6**(2), 461–464 (1978)
14. Yang, Z.R., Zwolinski, M.: Mutual information theory for adaptive mixture models. IEEE Trans. Pattern Anal. Mach. Intell. **23**(4), 396–403 (2001)
15. Bozdogan, H.: A new class of information complexity (ICOMP) criteria with an application to customer profiling and segmentation. Istanbul Univ. J. Sch. Bus. Adm. **39**(2), 370–398 (2010)
16. Pekhovsky, T., Lokhanova, A.: Variational Bayesian model selection for GMM-speaker verification using universal background model. In: INTERSPEECH 2011, 12th Annual Conference of the International Speech Communication Association, Florence, Italy, pp. 2705–2708, 27–31 August 2011 (2011)
17. Smyth, P.: Model selection for probabilistic clustering using cross-validated likelihood. Stat. Comput. **10**(1), 63–72 (2000)
18. Lee, Y., Lee, K.Y., Lee, J.: The estimating optimal number of Gaussian mixtures based on incremental k-means for speaker identification. Int. J. Inf. Technol. **12**(7), 13–21 (2006)
19. Ayoub, B., Jamal, K., Arsalane, Z.: An analysis and comparative evaluation of MFCC variants for speaker identification over VoIP networks. In: 2015 World Congress on Information Technology and Computer Applications Congress (WCITCA), pp. 1–6 (2015)

Sarcasm Identification on Twitter: A Machine Learning Approach

Aytuğ Onan(✉)

Department of Software Engineering, Faculty of Technology,
Celal Bayar University, 45400 Manisa, Turkey
aytug.onan@cbu.edu.tr

Abstract. In recent years, the remarkable growth in social media and microblogging platforms provide an essential source of information to identify subjective information of people, such as opinions, sentiments and attitudes. Sentiment analysis is the process of identifying subjective information from source materials towards an entity. Much of the social content online contain nonliteral language, such as irony and sarcasm, which may degrade the performance of sentiment classification schemes. In sarcastic text, the expressed text utterances and the intention of the person employing sarcasm can be completely opposite. In this paper, we present a machine learning approach to sarcasm identification. In this scheme, we utilized lexical, pragmatic, dictionary based and part of speech features. We employed two kinds of features to describe lexical information: unigrams and bigrams. In addition, term-frequency, term-presence and TF-IDF based representations are evaluated. To evaluate predictive performance of different representation schemes, Naïve Bayes, support vector machines, logistic regression and k-nearest neighbor classifiers are utilized.

Keywords: Sarcasm identification · Twitter · Machine learning

1 Introduction

Automatic identification of sarcasm is an important problem in natural language processing. With the advances in World-Wide Web (WWW), there is a remarkable growth in social media and microblogging platforms. Hence, a large amount of information is available on the web. This information can serve as an important source to identify subjective information of people, such as opinions, sentiments and attitudes.

Sentiment analysis (also known as opinion mining) is the process of identifying subjective information in source materials. The identification of public sentiment toward policies, products or services can be beneficial to the organizations [1]. Being able to identify subjective information is very important to generate structured knowledge that will serve crucial information to decision support systems and individual decision makers [2]. Much of the social content available on the Web contain nonliteral language, such as irony and sarcasm. For instance, the Internet Argumentation Corpus collected from 4forums.com contains utterances, 12% of which has sarcasm [3]. Sarcastic languages can degrade the predictive performance of sentiment

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_37

classification schemes. In sarcastic text documents, the expressed text utterances and the intention of the person employing sarcasm can be completely opposite.

Automatic identification of sarcasm is in its infancy [4]. One reason is that sarcasm is a hard concept to define. Since sarcasm is an ambiguous concept, it is even difficult for people to precisely identify whether a sentence is sarcastic or not [5]. Another reason is the absence of accurately-labeled naturally occurring utterances labeled as sarcastic that can be used to train supervised learning algorithms [4, 5]. In microblogging platforms, such as Twitter, users can express their opinions, feelings and ideas in short messages called tweets, within 140-character limit. Twitter is a fact growing microblogging platform with over 310 million monthly active users as of June 2016 [6]. Twitter enables users to communicate in a faster mode. Users can use Twitter for several purposes, such as daily chatter, conversation, sharing information and reading breaking news [7]. Recently, Twitter serves as an important source of information for researchers and practitioners, owing to the abundant amount of user-generated text messages on Twitter [8].

In this paper, we present a machine learning approach to identify sarcasm on Twitter messages. We report empirical results of the use of lexical, pragmatic, dictionary based and part of speech features in sarcasm identification. We employed two kinds of features to describe lexical information: unigrams and bigrams. In addition, term-frequency, term-presence and TF-IDF based representations are taken into consideration. In the classification phase, the predictive performance of four supervised learning algorithms (namely Naïve Bayes algorithm, logistic regression algorithm, support vector machines and K-nearest neighbor algorithm) are examined.

The rest of the paper is structured as follows: In Sect. 2, related work on sarcasm identification. Section 3 presents the methodology of the study. In Sect. 4, experimental procedure and empirical results are presented. Finally, Sect. 5 presents the concluding remarks.

2 Related Work

This section briefly reviews the existing works on automated identification of sarcasm. There are many works dedicated to sarcasm and irony in the literature of linguistics, psychology and cognitive science [4, 9–11].

In the domain of text mining, automatic identification sarcasm is considered as a challenging problem [4]. Tepperman et al. [12] examined the predictive performance of prosodic, spectral and contextual features on automatic identification of sarcasm. The empirical analysis indicated that the utilization of contextual features in conjunction with spectral features produces the highest predictive performance in terms of F-measure and classification accuracy. In another study, Kreuz and Gaucci [13] examined the effect of lexical features (such as the use of certain parts of speech tags, punctuation marks, presence of interjections) on automated identification of sarcastic language. Davidov et al. [14] presented a semi-supervised classification scheme to identify sarcastic and non-sarcastic utterances from Twitter messages and product reviews. In another study, Gonzalez-Ibanez et al. [4] presented a machine learning approach to sarcasm identification. In this scheme, unigrams, the presence of dictionary-based lexical and pragmatic factors and the frequency of dictionary-based lexical and pragmatic factors were considered as features and support vector machines and logistic regression algorithms were taken as the supervised learning algorithms. In another study, Veale and Hao [15] presented a rule-based scheme to identify whether a given simile is sarcastic or not. In this scheme, Google search was employed to identify how likely a simile is. In another study, Riloff et al. [16] presented an iterative algorithm that can automatically learn phrases corresponding to positive sentiments and negative situations. The process initiates with a single seed word and a large set of sarcastic tweets. In this scheme, discriminative phrases are learned and the algorithm utilizes the learned list of sentiment and phrases in identification of sarcasm on newer tweets. In another study, Liebrecht et al. [17] examined the use of intensifiers and exclamations in identifying sarcasm on Twitter. They hypothesized that explicit markers (such as hashtags) are digital equivalents of nonverbal expressions indicating sarcasm in live interactions. More recently, Rajadesingan et al. [18] incorporated author-specific contextual information into account in the sarcasm identification. In this scheme, contextual features (such as user's familiarity with twitter, language and sarcasm) are examined. In another study, Bamman and Smith [19] examined the use of extra-linguistic contextual information on sarcasm identification. The empirical analysis indicated that the use of contextual utterance on Twitter, such as properties of author, the audience and immediate communicative environment, enhances the predictive performance of machine learning based classification schemes.

3 The Methodology

This section presents the data set collection, feature engineering utilized to represent sarcasm dataset and classification algorithms utilized in the empirical analysis.

3.1 Dataset Collection

In the dataset collection, we adopted the framework presented in [4]. To build the sarcasm dataset with sarcastic, positive and negative tweets, self-annotated tweets by Twitter users are utilized. Twitter messages with hash tags of "#sarcasm" or "sarcastic" are regarded as sarcastic tweets, whereas hash tags related to positive sentiments are regarded as positive tweets and hash tags related to negative sentiments are regarded as negative sentiments. We used Twitter API to collect tweets. In this way, we have a collection of 5000 sarcastic, 5000 positive and 5000 negative tweets. In order to preprocess the dataset, special characters, such as "@" and "#" are removed from the dataset. In addition, words related to class hash-tags are removed.

3.2 Feature Engineering

In this section, we examine the different feature engineering schemes on the identification of sarcastic utterances. In this scheme, lexical, pragmatic, dictionary-based, part of speech based features are utilized.

Lexical features: We used two kinds of lexical features to represent tweets, namely, unigrams and bigrams are considered. In the vector space model, the frequency of features are taken into consideration to represent text documents. For a particular word *t,* term frequency of *t* in document *d* is defined as *TF*(*t, d*). In addition to term frequency, term presence *TP*(*t, d*) may also be considered to represent features. In this scheme, presence or absence of a word is utilized such that each word *t* is represented by 1 if it is present on a given document *d* and 0, otherwise. Term scoring schemes (such as TF-IDF, term frequency-inverse document frequency) can also be employed to evaluate the importance of a particular word on a given document collection or corpus. Hence, we have utilized term frequency (*TF*), term presence (*TP*) and term frequency-inverse document frequency (*TF-IDF*) to represent tweets.

Pragmatic features: We used three pragmatic features to represent tweets. First, the presence of positive emoticons (such as smileys) is regarded as binary features. Second, the presence of negative emoticons (such as frowning faces) is regarded as binary features. In addition, the presence of words in the interjections list is regarded as a binary feature.

Dictionary-based features: We used NRC word-emotion association lexicon (also known as EmoLex) to derive the dictionary-based features [20, 21]. EmoLex lexicon consists of a list of English words and their associations with eight basic emotions (such as anger, fear, anticipation, trust, surprise, sadness, joy and disgust) and two sentiments (negative and positive). The annotations were manually handled by crowdsourcing on Mechanical Turk. The lexicon contains 14,182 unigrams and approximately 25,000 senses.

Part-of-speech based features: We used the number of words representing positive emoticons and the number of words representing negative emoticons as features. In this scheme, the number of positive emotions, the number of negative emotions and total number of emotions in each tweet are regarded as features.

3.3 Classification Algorithms

In the classification, the predictive performance of four supervised learning algorithms are evaluated. This section briefly explains the algorithms employed in empirical analysis.

Naïve Bayes algorithm (NB) is a probabilistic classification algorithm, which is based on Bayes' theorem [22]. Naïve Bayes algorithm has a simple structure, owing to its assumption of conditional independence. In this way, the required computations are simplified, while obtaining promising predictive performance comparable to other conventional supervised learning algorithms, such as decision trees and artificial neural networks.

Support vector machines (SVM) are supervised learning algorithms that can be employed to solve classification and regression problems. SVM can effectively classify both linear and non-linear data [23]. In SVM, a non-linear matching is employed to transform the original dataset into a higher dimensional hyper-plane. This hyper-plane is used to identify optimal decision boundary that partitions the data into the appropriate classes.

Logistic regression (LR) is a linear classification algorithm, which models the probability of events' occurrence as a linear function of a set of predictor variables [24]. In logistic regression, the decision boundaries are determined based on a linear function of the features. LR aims to optimize the likelihood function to identify class labels for documents. The parameters of LR is chosen so that the conditional likelihood is maximized [25].

K-nearest neighbor algorithm (KNN) is an instance-based classification algorithm that can be employed for classification and regression problems. In KNN, the class label for a particular instance is identified based on the *k*-nearest training instances of a particular instance. The majority voting is employed to combine the predictions of the neighbors of an instance. In this scheme, each instance is assigned to the majority vote of its neighbors, namely, the most common class among its *k*-nearest neighbors [26].

4 Experimental Analysis

This section presents evaluation metrics, experimental procedure and experimental results of empirical analysis.

4.1 Evaluation Measures

In order to evaluate the performance of classification algorithms, two different evaluation measures, namely, classification accuracy and F-measure.

Classification accuracy (ACC) is the proportion of true positives and true negatives obtained by the classification algorithm over the total number of instances as given by Eq. 1:

$$ACC = \frac{TN + TP}{TP + FP + FN + TN} \tag{1}$$

where TN denotes number of true negatives, TP denotes number of true positives, FP denotes number of false positives and FN denotes number of false negatives.

Precision (PRE) is the proportion of the true positives against the true positives and false positives as given by Eq. 2:

$$PRE = \frac{TP}{TP + FP} \tag{2}$$

Recall (REC) is the proportion of the true positives against the true positives and false negatives as given by Eq. 3:

$$REC = \frac{TP}{TP + FN} \tag{3}$$

F-measure takes values between 0 and 1. It is the harmonic mean of precision and recall as determined by Eq. 4:

$$F - measure = \frac{2 * PRE * REC}{PRE + REC} \tag{4}$$

4.2 Experimental Procedure

In the experimental analysis, 10-fold cross validation method is employed. In this scheme, the original dataset is randomly divided into ten mutually exclusive folds. Training and testing process is repeated ten times and each part is tested and trained ten times and the average results for 10-fold are reported. The experimental analysis is performed with the machine learning toolkit WEKA (Waikato Environment for Knowledge Analysis) version 3.9, which is an open-source platform that contains many machine learning algorithms implemented in JAVA. In the empirical analysis, we have performed a three-way comparison of sarcastic (S), positive (P) and negative (N) messages (denoted by PNS). In addition, we have performed three two-way comparisons of sarcastic (S), positive (P) and negative messages (N): namely, negative vs sarcasm classifier (NS), positive vs sarcasm classifier (PS) and non-sarcastic and sarcastic classifier (NSS) are also evaluated.

4.3 Experimental Results

In Tables 1 and 2, classification accuracies and F-measure values obtained by the compared algorithms on sarcasm identification are presented. As it can be observed from the experimental results presented in Tables 1 and 2, the highest predictive performance is obtained from three-way and two-way classification analysis, when term-presence and unigram features are utilized to represent text documents. Compared to bigram features, unigram features generally yield more promising results in terms of classification accuracies. In addition, term presence based representation yields more promising results compared to two other schemes, namely term-frequency based representation and TF-IDF based weighting scheme. Regarding the predictive performance of supervised machine learning algorithms, support vector machines and logistic regression classifiers outperform the other supervised learning algorithms. For negative vs sarcastic classifier (NS), three-way comparison of sarcastic, positive and negative messages (PNS), positive vs sarcasm classifier (PS), the highest classification accuracies are obtained by logistic regression algorithm. For non-sarcastic and sarcastic classifier (NSS), support vector machines yield the highest classification accuracy (89.15%).

Table 1. Classification accuracies obtained by supervised learning algorithms

	Representation	NB	SVM	LR	KNN
NS	TF, unigram	71.55	79.57	80.23	69.63
	TP, unigram	**71.89**	**81.2**	**81.83**	**70.15**
	TF-IDF, unigram	71.82	79.66	80.53	69.97
	TF, bigram	69.07	71.49	72.45	67.71
	TP, bigram	69.54	73.24	73.93	68.36
	TF-IDF, bigram	69.38	71.64	72.75	68.19
PNS	TF, unigram	72.92	73.77	77.64	72.88
	TP, unigram	**73.82**	**77.98**	**78.49**	**73.46**
	TF-IDF, unigram	73.72	74.55	78.38	72.94
	TF, bigram	71.61	72.65	73.36	71.95
	TP, bigram	72.20	73.73	74.39	72.47
	TF-IDF, bigram	71.94	72.69	73.33	72.30
PS	TF, unigram	72.09	78.25	79.29	69.38
	TP, unigram	**73.40**	**80.6**	**81.13**	**69.90**
	TF-IDF, unigram	72.27	78.35	79.62	69.80
	TF, bigram	69.25	71.15	72.24	67.37
	TP, bigram	72.03	73.01	73.47	67.84
	TF-IDF, bigram	69.96	71.33	72.59	67.76
NSS	TF, unigram	72.94	86.38	86.38	86.00
	TP, unigram	**77.14**	**89.15**	**88.87**	**87.12**
	TF-IDF, unigram	75.78	86.71	87.51	86.14
	TF, bigram	74.00	84.05	83.14	84.48
	TP, bigram	75.13	85.67	86.16	86.24
	TF-IDF, bigram	74.21	84.29	85.72	85.31

In Fig. 1, the comparisons of different representation schemes and supervised learning algorithms are provided. In Table 2, F-measure values obtained from the supervised learning algorithms on sarcasm identification are presented. Similar to the empirical results presented in Table 1, the best F-measure results are obtained by term presence and unigram based feature representation. In addition, support vector machines and logistic regression algorithms generally outperform the other supervised learning algorithms in terms of F-measure values. The results presented in empirical analysis indicate that lexical, pragmatic, dictionary-based and part-of speech based features utilized to represent tweets can yield promising results on sarcasm identification. The identification of an appropriate feature set is an important issue in developing robust machine learning based classification schemes. Hence, the experimental results presented in this section may be further enhanced with the use of contextual features and more other feature engineering schemes.

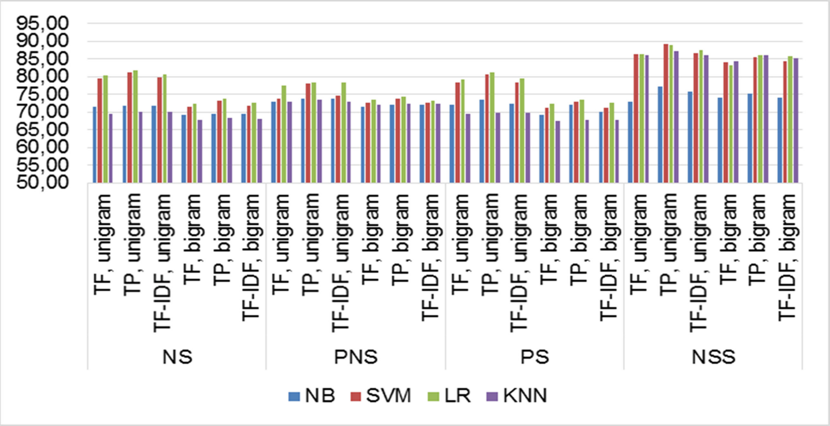


Fig. 1. Average classification rates for algorithms

Table 2. F-measure values by supervised learning algorithms

	Representation	NB	SVM	LR	KNN
NS	TF, unigram	0.81	0.85	0.84	0.75
	TP, unigram	**0.85**	**0.87**	**0.85**	**0.78**
	TF-IDF, unigram	0.84	0.85	0.84	0.77
	TF, bigram	0.77	0.74	0.74	0.70
	TP, bigram	0.80	0.76	0.76	0.74
	TF-IDF, bigram	0.80	0.75	0.74	0.74
PNS	TF, unigram	0.80	0.80	0.79	0.73
	TP, unigram	**0.83**	**0.81**	**0.84**	**0.74**
	TF-IDF, unigram	0.81	0.80	0.80	0.74
	TF, bigram	0.72	0.78	0.78	0.65
	TP, bigram	0.80	0.80	0.78	0.71
	TF-IDF, bigram	0.77	0.79	0.78	0.67
PS	TF, unigram	0.71	0.74	0.73	0.59
	TP, unigram	**0.79**	**0.76**	**0.83**	**0.66**
	TF-IDF, unigram	0.78	0.74	0.73	0.63
	TF, bigram	0.64	0.64	0.63	0.55
	TP, bigram	0.67	0.66	0.64	0.58
	TF-IDF, bigram	0.66	0.64	0.63	0.56
NSS	TF, unigram	0.80	0.86	0.85	0.81
	TP, unigram	**0.86**	**0.89**	**0.86**	**0.86**
	TF-IDF, unigram	0.84	0.87	0.84	0.84
	TF, bigram	0.78	0.85	0.83	0.74
	TP, bigram	0.80	0.86	0.85	0.77
	TF-IDF, bigram	0.78	0.85	0.84	0.76

5 Conclusion

With the advances in information and communication technologies, there is a remarkable growth in social media and microblogging platforms. Microblogging platforms can serve as an important source of information for identifying subjective information of people, such as opinions, attitudes and sentiments. Automatic identification of sarcasm is a challenging problem in natural language processing. In this paper, we have presented a machine learning based approach to identify sarcasm on Twitter messages. We have examined the predictive performance of different representation schemes, such as term-frequency, term-presence and TF-IDF based representations. In addition, two kinds of features (namely, unigrams and bigrams) are utilized to describe lexical information. In the classification phase, the predictive performance of four supervised learning algorithms (namely Naïve Bayes algorithm, logistic regression algorithm, support vector machines and K-nearest neighbor algorithm) are examined. Regarding the empirical results, the highest predictive performance (89.15%) is obtained by term-presence and unigram features, when support vector machines are utilized as the supervised learning algorithm.

References

1. Wang, G., Sun, J., Ma, J., Xu, K., Gu, J.: Sentiment classification: the contribution of ensemble learning. Decis. Support Syst. **57**, 77–93 (2014)
2. Fersini, E., Messina, E., Pozzi, F.A.: Sentiment analysis: Bayesian ensemble learning. Decis. Support Syst. **68**, 26–38 (2014)
3. Walker, M.A., Tree, J.E.F., Anand, P., Abbott, R., King, J.: A corpus for research on deliberation and debate. In: Proceedings of Language Resources and Evaluation Conference, pp. 812–817. ACL, New York (2012)
4. Gonzalez-Ibanez, R., Muresan, S., Wacholder, N.: Identifying sarcasm in Twitter: a closer look. In: Proceedings of the 49th Annual Meeting of the Association for Computation Linguistics, pp. 581–586. ACL, New York (2011)
5. Muresan, S., Gonzalez-Ibanez, R., Ghosh, D., Wacholder, N.: Identification of nonliteral language in social media: a case study on sarcasm. J. Assoc. Inf. Sci. Technol. (2016). doi:10.1002/asi.23624
6. About.twitter.com: Company | About. (2016). https://about.twitter.com/company. Accessed 15 Jan 2017
7. Java, A., Song, X., Finin, T., Tseng, B.: Why we Twitter: understanding microblogging usage and communities. In: Proceedings of the 9th WebKDD Conference, pp. 56–65. ACM, New York (2007)
8. Onan, A.: A machine learning approach to identify geo-location of Twitter users. In: Proceedings of the Second International Conference on Internet of Things, Data and Cloud Computing (2017)
9. Gibbs, R.: On the psycholinguistic of sarcasm. J. Exp. Psychol. **105**, 3–15 (1986)
10. Gibbs, R., Colston, H.L.: Irony in Language and Thought. Taylor and Francis, New York (2007)
11. Utsumi, A.: Verbal irony as implicit display of ironic environment: distinguishing ironic utterances from nonirony. J. Pragm. **32**(12), 1777–1806 (2000)

12. Tepperman, J., Traum, D.R., Narayanan, S.: "yeah right": sarcasm recognition for spoken dialogue systems. In: Proceedings of the Ninth International Conference on Spoken Language Processing, pp. 1–4. Carnegie Mellon University, Pittsburgh (2006)
13. Kreuz, R.J., Gaucci, G.M.: Lexical influences on the perception of sarcasm. In: Proceedings of the Workshop on Computational Approaches to Figurative Language, pp. 1–4. ACM, New York (2007)
14. Davidov, D., Tsur, O., Rappoport, A.: Semi-supervised recognition of sarcastic sentences in Twitter and Amazon. In: Proceedings of the Fourteenth Conference on Computational Natural Language Engineering, pp. 107–116. ACM, New York (2010)
15. Veale, T., Hao, Y.: Detecting ironic intent in creative comparisons. In: ECAI, pp. 765–770. IOS Press, Amsterdam (2010)
16. Riloff, E., Qadir, A., Surve, P., De Silva, L., Gilbert, N., Huang, R.: Sarcasm as contrast between a positive sentiment and negative situation. In: EMNLP, pp. 704–714. ACM, New York (2013)
17. Liebrecht, C.C., Kunneman, F.A., van Den Bosch, A.P.J.: The perfect solution for detecting sarcasm in Tweets# not. In: Proceedings of the 4th Workshop on Computational Approach to Subjectivity, Sentiment and Social Media Analysis, pp. 29–37. ACL, New York (2013)
18. Rajadesingan, A., Zafarani, R., Liu, H.: Sarcasm detection on Twitter: a behavioral modeling approach. In: Proceedings of the Eight ACM International Conference on Web Search and Data Mining, pp. 97–106. ACM, New York (2015)
19. Bamman, D., Smith, N.A.: Contextualized sarcasm detection on Twitter. In: Ninth International AAAI Conference on Web and Social Media, pp. 574–577. AAAI Press, New York (2015)
20. Mohammad, S., Turney, P.: Crowdsourcing a word-emotion association lexicon. Comput. Intell. **29**(3), 435–465 (2013)
21. Mohammad, S., Turney, P.: Emotions evoked by common words and phrases: using mechanical turk to create an emotion lexicon. In: Proceedings of the NAACL-HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, pp. 26–35. ACL, New York (2010)
22. John, G.H., Langley, P.: Estimating continuous distributions in Bayesian classifiers. In: Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, pp. 338–345. Morgan Kaufmann, San Francisco (1995)
23. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York (1995)
24. Kantardzic, M.: Data Mining: Concepts, Models, Methods and Algorithms. Wiley, New York (2011)
25. Aggarwal, C.C., Zhai, C.X.: A survey of text classification algorithms. In: Aggarwal, C.C., Zhai, C.X. (eds.) Data Mining and Knowledge Discovery Handbook, pp. 77–128. Springer, Berlin (2012)
26. Han, J., Kamber, M., Pei, J.: Data Mining: Concepts and Techniques. Morgan Kaufmann, San Francisco (2011)

The Problem of Critical Events' Combinations in Air Transportation Systems

Leonid Filimonyuk(✉)

Institute of Precision Mechanics and Control of Russian Academy of Sciences,
Saratov, Russia
rezchikov1939@mail.ru

Abstract. The approaches to analysis and classification of causes of accidents in air transportation systems are offered in the article. Accidents and catastrophes are represented as consequences of critical combinations of harmful events that caused by an instruction's breach, a technical failure and outside forces on an airplane. Offered approaches can be used for development and modification of training systems and simulators for pilots and air-traffic controllers.

Keywords: Air transportation system · Safety · Accidents · Combination of events · Informational and logical diagram

1 Introduction

The problem of the security of air transportation systems [1] is very important due to serious consequences of accidents. A unified control of air transportation system does not exist in Russia that is why this research is actual. There are a few articles devoted to the problem [1–5, 9–14].

The solution of the problem requires research of large-scale and complicated systems that generally do not have strong mathematical description. The most important step in this case is to use the cause-effect approach [1], and methods for analyzing the interaction of complex systems resources including human factor [5–7].

Aircraft accidents and their ground effects consider as the development of negative events' chains connected by temporal, cause-effect and other relations. In many cases the preconditions of these events chains are the instruction breakings that lead the system to a situation of lack of system resources (time, skills, energy, information) to prevent extreme external influences, or a combination of several subsystems failures. Breaches of the instructions may have a regular disposition, but it becomes the accidents' cause only in the case of critical events' combination.

The events' chain leading to the accident can be interrupted at various stages and each of them requires its own set of tools to use with the corresponding resources. An accident happens when prevention doesn't occur on any of these stages.

This article proposes an approach to the classification and analysis of the critical events combination as the causes of accidents.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_38

2 The Safety of Air Transportation Systems and the Problem of Critical Events' Combinations

The elements comprising this system are represented on Fig. 1.

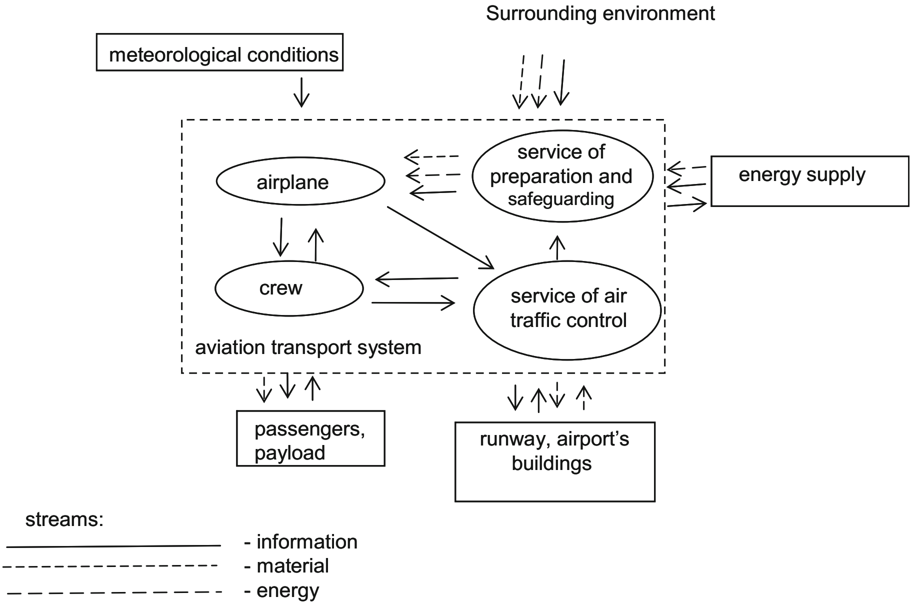

Surrounding environment
meteorological conditions
airplane
service of preparation and safeguarding
energy supply
crew
service of air traffic control
aviation transport system
passengers, payload
runway, airport's buildings
streams:
- information
- material
- energy

Fig. 1. An air transportation system's structure

Flight safety is defined as an air transportation system's property that describes the level of danger to people and material objects onboard the aircraft in the process of its functioning.

A model consisting of two interconnected units is proposing for the problem solution: the mathematical models complex of the aircraft dynamics and cause-effect models as a set of conditions describing the normal functioning of the air transportation system, including actions of the crew and air traffic controllers.

Analysis of occurred accidents shows, that they are caused by complex of events' combinations, each of which occurs due to the certain resources lack.

Negative events may be a result of endogenous factors or external influences, and form a cause-effect chain.

The crew or air traffic controller error also belongs to the subsystems failures category [2, 8–11]. A lot of attention in various studies paid for preventing single failures and errors [3, 4, 6]. Models and methods of research and prevent combinations of heterogeneous subsystems are less developed. This problem is particularly acute when safety depends on the decisive actions of the crew and the air traffic controller. Emergency situation requires faultless and operational control actions for its prevention. The right decision making resources are limited by psychophysiological characteristics and the crew's experience. If prevention does not occur, the situation is growing more rapidly, while prevention capabilities reduce.

3 An Approach to Classification of Critical Combination of Events

By the critical combination of events we consider the situation when negative events coming coincide with means of prevention failure.

For increase of the aviation safety classification of the air transportation system's critical events combination is necessary. The following directions of the events' combination classification in air transportation systems are offered:

- by the time of occurrence relatively to the accident: before the crash, during the crash, during the prevention;
- by timing characteristics of these events combination;
- by the intensity of the appearance of this type of events combinations at this point in time, by the frequency of occurrence;
- by objects and elements according to their layering defined by the technical documentation [4]: management personnel; hardware and software control systems; technical part of the air transportation system; the energy part of air transportation system; payload and passengers; environment and further by the elements of the subsystems;
- by composition and quantity of diverse processes involved in combined events: processes of crews and air traffic controllers actions; control processes; air transportation system's subsystems and units functioning process; fuel and energy processes; processes of passengers actions, movement of payload, etc.; The interaction with the environment processes;
- by type of the crew and air traffic controllers errors: organizational errors - negligence, deliberate or accidental misrepresentation; human error in the interaction with the technical subsystem;
- according to the individual danger degree of failures forming the combination under consideration: a combination of non-critical failures; coupled with critical failures and errors.
- by the implementation difficulty and resource consumption for preventing and countering the critical situation at the moment. It means that in the initial stages of failures sequence difficulty of prevention can be significantly lower, and efficiency can be higher.
- The classification lets develop methods for predicting the critical events' combination.

4 Accidents Description and the Conditions of Their Occurrence

Formal mathematical representation of the critical events combination is needed to make use of computers,

Consider the general scheme of critical events combination. The mathematical logic and set theory can be used to describe the scheme. Through $P(a_1, t_1)$ and $\overline{P}(a_1, t_1)$ denote the staffing and harmful events. The last means a breaking of the security conditions and the occurrence of failure at time t1, associated with a subsystem a_1. This event may imply that in case of necessity of the further subsystem functioning:

- value $r_1(t_1)$ of resource indicator a_1 is less than critical: $r_1(t_1) < \underline{\underline{r}}_1$;
- value $r_1(t_1)$ of resource indicator a_1, if the indicator represented as list, doesn't contains required component: $\underline{\underline{r}}_1 \not\subset r_1(t_1)$, where $\underline{\underline{r}}_1$ – is a resource list necessary for the regular operation of the component subsystem a_1.

The equation of the correct air transportation system's functioning is

$$\left(\forall t \in [t_1, t_1 + \delta_p]\right) P(a_1, t_1) \wedge P(a_2, t) \wedge \ldots \wedge P(a_n, t) \rightarrow \ll \text{correct functioning} \gg$$

i.e. safety conditions applies for all moments of system's functioning.

Suppose that in order to avoid the development of a critical situation relating to this harmful event, it should be prevented for time not larger than δ_p. Assume that each of the subsystems a_j, j = 2, …, n with interchangeable and sufficient toward a_1 resources can accomplish harmful events' prevention related to a_1 for time less than δ_p. Condition of other subsystem a_j failure occurrence event in moment t denote as $\overline{P}(a_j, t)$, where j = 2, …, n. In this case, the following condition must be satisfied to prevent the development of an emergency: $P(a_2, t) \vee \ldots \vee P(a_n, t)$ at $t \in [t_1, t_1 + \delta_p]$. That is, at least one of the subsystems a_j has sufficient resources to compensate for lack of resources a_1. Then the condition of the emergency functioning is of the form:

$$\left(\forall t \in [t_1, t_1 + \delta_p]\right) \overline{P}(a_1, t_1) \wedge \overline{P}(a_2, t) \wedge \ldots \wedge \overline{P}(a_n, t) \rightarrow \ll \text{accident} \gg$$

This means that subsystem a_1 lack of resources и and simultaneous safety conditions non-compliance for a$_1$ and other subsystems a_j, j = 2, …, n on the considered time interval leads to an accident as other subsystems cannot compensate a_1 lack of resources.

It is possible to describe the critical events combination with required level of decomposition in the fairly complicated system using computer, as logical security conditions can be easily programmed.

5 Informational and Logical Diagram for the Air Transportation System Control in a Case of Critical Events' Combination

According to [1] in order to solve the stated problem, it's necessary:

- to work out a list of air transportation system's events $\{D_1, D_2, \ldots, D_m\}$ according to the classification;
- to build an event tree *D*, which combines trees $\{D_1, D_2, \ldots, D_m\}$ and allows calculation of critical combinations of events in the air transportation system;
- to work out a mathematical model allowing numerical calculation of probabilities of different critical combinations.

During the aircraft flights monitoring of appearing events is carried out. The events correspond to various nodes of the *D* tree, and their combinations cause emergency situations. When event *a* takes place, which is classified according to the list of accidents and catastrophes as an adverse one, a selection is made of the $\mathrm{E}(a)$ set of all the critical combinations of events containing event *a,* and the probabilities of all these events are calculated. The $\mathrm{E}_1(a) \subseteq E(a)$ set of critical combinations of events is determined, whose probabilities exceed the known threshold values. The system selects formed in advance lists of actions for preventing adverse events from the database, which allows prevention of critical combination of events.

The normative time for elimination of the malfunction while realizing each list of actions is known, that is, restoration intensities $\mu_{\mathrm{j}}(t)$, $j = \overline{1, m}$ of functioning of the failed elements of the air transportation system are known.

When it's impossible for the crew members and flying control officers to fulfil their duties and in the absence in the control system of the list of actions, which would allow the elimination of the situation, the crew and the flying control officer actively form the corresponding lists of actions and estimate the time of their realization, i.e. calculate the $\mu_{\mathrm{j}}(t)$, $j = \overline{1, m}$ values.

Part of this list of actions is not carried out automatically and is meant for pilots and flying control officers. This part of the list contains several versions of clear and suitable for realization instructions, application of which depends on the decision made by DMP (decision-making person) in the circumstances. The informational and logical diagram of the problem solution is shown on Fig. 2.

The following designations are assumed for Fig. 2: 1, 2 – parameters describing the process of the aircraft's functioning and the state of the crew, respectively; 3, 4 – parameters of the air transportation system, defined by the air transportation control and flight preparation personnel, respectively; 5 – collection of information about the process of the air transportation system's functioning, coming from sensors and local automated devices; 6 – entering information into a database or into a knowledge base; 7 – estimation of the situation when parameters of the process of the air transportation system's functioning deviate significantly from the preset values; 8 – identification of the current events affecting the safety of the functioning air transportation system; 9 – determining the list of the minimal sections including events from item 8; 10 – calculation of the probability of critical combinations of events from item 9; 11 – do

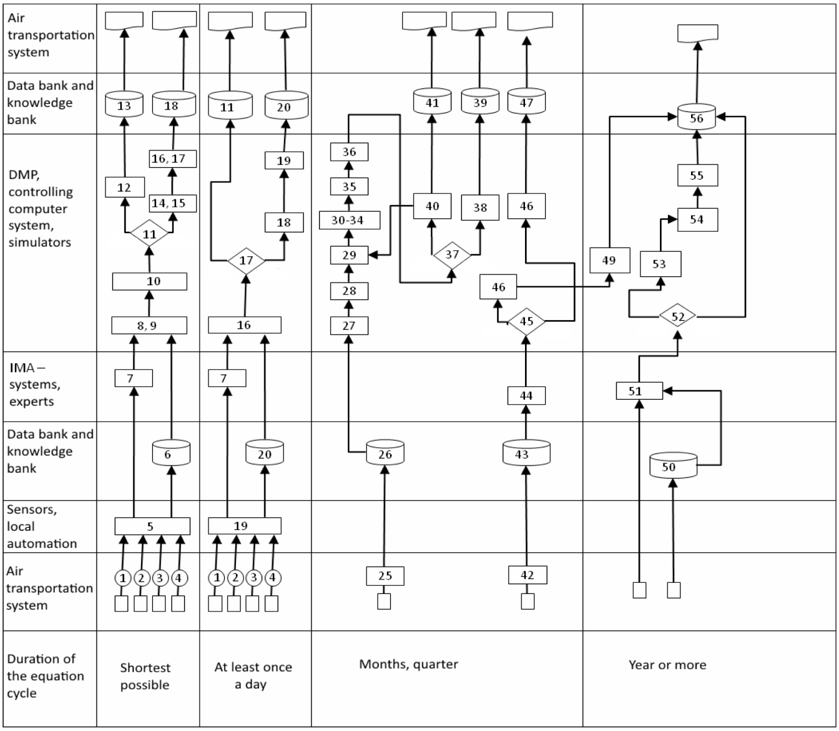

Air transportation system
Data bank and knowledge bank
DMP, controlling computer system, simulators
IMA – systems, experts
Data bank and knowledge bank
Sensors, local automation
Air transportation system
Duration of the equation cycle
Shortest possible
At least once a day
Months, quarter
Year or more
13
18
16, 17
12
14, 15
11
10
8, 9
7
6
5
1
2
3
4
11
20
19
18
17
16
7
20
19
1
2
3
4
41
39
47
36
35
30-34
29
28
27
40
38
46
37
46
45
44
43
26
25
42
56
55
54
49
53
52
51
50

Fig. 2. Informational and logical diagram for the air transportation system control in a case of critical events' combination

probabilities of some critical combination of events exceed acceptable thresholds?; 12 – the message about the fact that the probabilities of the considered combinations of events do not exceed the acceptable thresholds; 13 – entering information about non-dangerous event combinations to the server of the air transportation control service; 14 – the message about the fact that the probabilities of the considered combinations of events exceed the acceptable thresholds; 15 – building a radar chart reflecting probabilities of emergency situation when different event combinations take place; 16 – determining of the vector of the optimum values $\mu_i(t)$, $i = \overline{1, n}$ and corresponding actions aimed at eliminating reasons for critical combinations of events; 17 – accepting and realizing solutions for eliminating critical combinations of events, which can cause emergencies; 18 – entering information about an emergency and measures taken for its elimination to the server of the air transportation control service; 19 – collecting information about the current state of the air transportation system's subsystems; 21 – selection of a controlled list of the critical combinations of events; 22 – displaying a message about the danger of a critical combination of events in the process of functioning of the *i-th* subsystem of the air transportation system; 23 – issuing a recommendation on actions aimed at eliminating the reasons of a probable emergency situation; 24 – entering information into the database; 25 – launching the simulators; 26 – calling an event tree D*, used for training crews and flying control officers to act when emergency situations arise because of critical combinations of events, from the database; 27 – determining minimal sections of the tree D* corresponding to various combinations of events; 28 – working out algorithms for various minimal sections; 29 – forming a block of test problems; 30 – building a radar chart reflecting probabilities of emergency situation when different event combinations take place; 31 – determining a list of actions aimed at eliminating of the critical combination of events, which are optimal according to safety criterion; 32 – questioning a trainee on which critical situations, in his opinion, it is necessary to take into account while making a decision (pick from a list); 33 – ranging the critical combinations of events that have appeared by the level of danger and estimating the danger of their appearance; 34 – forming by the trainee of the list of actions necessary to eliminate the reasons of the critical combination of events that has taken place; 35 – comparing the list of actions created by the trainee with the test list; 36 – estimation of the level of the trainee's preparedness to making adequate decisions when various critical combinations of events arise; 37 – was the test passed successfully?; 38 – reward; 39 – entering information into the database about successful completion of training; 40 – analysis of errors; 41 – entering information into the database; 42 – launching the procedure of collecting and analyzing statistical data about the emergencies that happened during the last month due to critical combination of events; 43 – determining repeatable emergencies; 44 – elimination of the reasons of repeatable critical combinations of events; 45 – was it successful?; 46 – issuing recommendations on eliminating the critical combination of events; 47 - entering information into the database; 48 – issuing recommendations on the air transportation system's structure changes; 49 – changing the air transportation system's structure; 50 – accumulating information about control modifications applied during the last year; 51 – expert estimation of the size of the economical effect from the application of the control modifications; 52 – was the expected economical effect of the problem solution reached or exceeded? 53 – analysis of reasons; 54 – correction of the plan of actions aimed at minimizing the damage from the emergencies caused by the critical combination of events; 55 – getting approval for the corrected plan of actions; 56 – entering information into the database.

For the minimal period of time (seconds, minutes, or hours), the problem of control is solved, and the work of the system is directed at supporting the decision making for crews and flying control officers during the flight. Information about critical combinations of defects and errors collected at this stage is systematized in data banks, processed and analyzed.

For the medium time period (day, month, or quarter), training of pilots and flying control officers on the basis of the collected information takes place, which allows avoiding emergency situations in the future. Information about training results and of the possibility of avoiding critical combinations of events is systematized and analyzed in the corresponding databases.

For long time periods accumulation of information and knowledge of known emergency situations takes place, and the problem of renovating of air transportation systems gets solved: alterations of the normative documents, improving the design of the aircraft. The solution of such problems is necessary to help managers at various levels, system architects, aircraft designers, etc.… When necessary, correction of parameters of the mathematical model is carried out to use for controlling an air transportation system with safety as the criterion. This correction can be especially important when the same failures are repeated often. The problem can be solved by means of correcting regulations, laws and rules, constant adaptation of the control system and accumulating information in data banks.

6 Conclusion

The main purposes of improving the air transportation systems safety are creating models of all parts of the air transportation system, the development of technology-based requirements for fail-safety, including crew's errors.

The approaches to the cause-effect description of the critical events combination that cause accidents in the air transportation system are developed. The proposed methods can be used to investigate and forecast the causes of accidents and catastrophes in air transportation systems.

The logical conditions for description of the most important causes of accidents difficult to prevent, as well as the further critical events combination classification development put on the basis of the proposed approach.

The proposed language for describing and formalizing is simple to use by engineers for programming and description of the big number of known accidents in the world, presenting them in a compact and convenient form for the formation and collecting in data bases.

The proposed approaches and research results could be used for simulators and decision-making systems and in air transportation systems. These developments are used to improve the model reliability and safety of air transportation systems created by Open Joint Stock Company «Ilyushin Aviation Complex».

Acknowledgments. The research is supported by Russian foundation for basic research, grant number 16-01-00536 a.

References

1. Rezchikov, A., Kushnikov, V., Ivaschenko, V., Bogomolov, A., Filimonyuk, L., Kachur, K.: Control of the air transportation system with flight safety as the criterion. Autom. Control Theory Perspect. Intell. Syst. **466**, 423–432 (2016)
2. Hansman, R.J., Histon, J.M.: Mitigating complexity in air traffic control: the role of structure-based abstractions. In: ICAT - Reports and Papers (5) (2008)
3. Vaaben, B., Larsen, J.: Mitigation of airspace congestion impact on airline networks. J. Air Transp. Manag. **47**, 54–65 (2015)
4. Alderson, J.C.: Air safety, language assessment policy and policy implementation: the case of aviation English. Annu. Rev. Appl. Linguist. **29**, 168–187 (2009)
5. Rezchikov, A., Dolinina, O., Kushnikov, V., Ivaschenko, V., Kachur, K., Bogomolov, A., Filimonyuk, L.: The problem of a human factor in aviation transport systems. Indian J. Sci. Technol. **9**(46) (2016)

6. Rezchikov, A., Dolinina, O., Kushnikov, V., Ivaschenko, V., Kachur, K., Bogomolov, A., Filimonyuk, L.: An approach to the development of the control system of the aviation transport safety taking into account the human factor. Int. J. Eng. Sci. Res. Technol. **6**(2), 279–284 (2017)
7. Rezchikov, A., Dolinina, O., Kushnikov, V., Ivaschenko, V., Kachur K., Bogomolov, A., Filimonyuk, L.: The problem of a human factor in aviation transport systems. In: Proceedings of the 3rd International Conference on Computing, Technology and Engineering, pp. 16–20 (2016)
8. Vossen, T.W.M., Hoffman, R., Mukherjee, A.: Quantitative Problem Solving Methods in the Airline Industry, pp. 385–447. Springer, New York (2012)
9. Aigoin, G.: Air traffic complexity modeling. Master thesis, ENAC (2001)
10. Kopardekar, P., Schwartz, A., Magyarits, S., Rhodes, J.: Airspace complexity measurement: an air traffic control simulation analysis. In: 7th USA/Europe Air Traffic Management R&D Seminar (2007)
11. Sollenberger, R., Koros, A., Hale, M.: En route information display system benefits study. In: William, J. (ed.) Hughes Technical Center (2008)
12. Marla, L., Vaaben, B., Barnhart, C.: Integrated disruption management and flight planning to trade off delays and fuel burn. Technical report. DTU Management Engineering (2011)
13. Altus, S.: Quantitative Problem Solving Methods in the Airline Industry, pp. 295–315. Springer, New York (2012)
14. Bertsimas, D., Lulli, G., Odoni, A.: An integer optimization approach to large-scale air traffic flow management. Oper. Res. **59**(1), 211–227 (2011)

Using Text Mining Methods for Analysis of Production Data in Automotive Industry

Lukas Hrcka[1,2(✉)], Veronika Simoncicova[1,2], Ondrej Tadanai[1,2],
Pavol Tanuska[1,2], and Pavel Vazan[1,2]

[1] Faculty of Materials Science and Technology,
Slovak University of Technology, Trnava, Slovak Republic
{lukas.hrcka, veronika.simoncicova, ondrej.tadanai,
pavol.tanuska, pavel.vazan}@stuba.sk
[2] Institute of Applied Informatics,
Automation and Mechatronics, Trnava, Slovak Republic

Abstract. Text mining is the process of extracting useful and high-quality information from unstructured textual data through the identification and exploration of interesting patterns. Text mining also referred to as text data mining, roughly equivalent to text analytics. RapidMiner is unquestionably the world-leading open-source system for this analytics, it is the most powerful and easy to use. Acquiring information from text is a requested area of research in automotive industry. This paper aims at presenting the use of text mining in this industry field. The article is focused on working with text attributes "ResponsibleEmp", which is crucial for text mining analysis. The outcome of this article is the number of breakdowns and name of a specific employee responsible for breakdowns. The presented analysis is provided as a partial result of the research and will serve to further investigation in the problem area.

Keywords: RapidMiner · Text mining · Analysis · Data · Information

1 Introduction

Text mining can help an organization to acquire potentially valuable business insight. Text mining from unstructured data might be a challenging task as natural language processing; statistical model and machine learning techniques are often required due to inconsistent syntax and semantics of natural language text, including slang and language specific to vertical industries [1].

Industries dispose of a large amount of production databases, where huge amounts of data are stored. A huge amount of company information (approximately 80%) is available in textual data formats, which is why the text mining is useful and important in industrial management [9].

The automotive industry is more data-driven today than at any time in its history. In-car sensors, GPS tracking, automated manufacturing processes, and more are producing vast volumes of data that need to be analysed and understood. RapidMiner's Predictive Analytics platform enables car makers to derive value from this data by extracting the information hidden online, inside the vehicle, or at the plant with the

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_39

purpose of better understanding product usage, preferences, and manufacturing processes to ensure quality and customer satisfaction [10].

Our research is based on production data in text documents, especially forms and the goal of our research is executing production data examination and further analysis in automotive industry using RapidMiner, an open source tool useful for text mining.

2 Text Mining and Solution Proposal

2.1 Knowledge Discovery and Methodology

Knowledge discovery in process automation represents extracting knowledge from different types of data, for example – KDD: Knowledge Discovery in Databases, KDT: Knowledge Discovery in Text or TM: Text Mining. Knowledge Discovery in Text is also called Text Mining. Due to the unstructured language, the process of knowledge discovery in a set of text documents seems much more complex than the process of knowledge discovery in databases [2].

A Text Mining Analysis of Academic Libraries' Tweets, a respected publication in the field, deals with clarifying the operation of text mining. The authors try to apply a text mining approach to a significant dataset of tweets by academic libraries and authors acquiring information about searching the most frequently one word, two-word and tree-word expressions and visiting academic libraries. These findings highlight the importance of using data and text mining approaches in understanding the aggregate social data of academic libraries to aid in decision-making and strategic planning for marketing of services [3].

Seo Wonchul researched how to extract product information from patent database using text mining technique and then generate product connection rules represented as directed pairs of products. Finally, authors evaluated potential value of product opportunities taking into account firm's internal capabilities [4]. Their approach can facilitate product-oriented research and development by presenting a front-end model for new product development and deriving feasible product opportunities according to the target firm's internal capabilities [4].

Knowledge discovery of text mining is a general process which in our research consists of several steps are shown in Fig. 1.

(1) Obtaining data and transformation into required format: The first step in this process is to create an input text file containing a list of attributes and values. Obtained unstructured data must be transformed into a structured CSV file format. Subsequently, the modified data are better processed and the work is efficient and more effective.

(2) Pre-processing: CSV file is used as the input data source for RapidMiner. This source data is loaded into RapidMiner and functions for editing the text and eliminating inconsistent or erroneous data are used. Subsequently, the values are retyped using "Process Documents from Data" operator, where methods for text editing as tokenization, filters, stop wording, etc. are used. Such modified data are ready to be processed with analytical techniques.

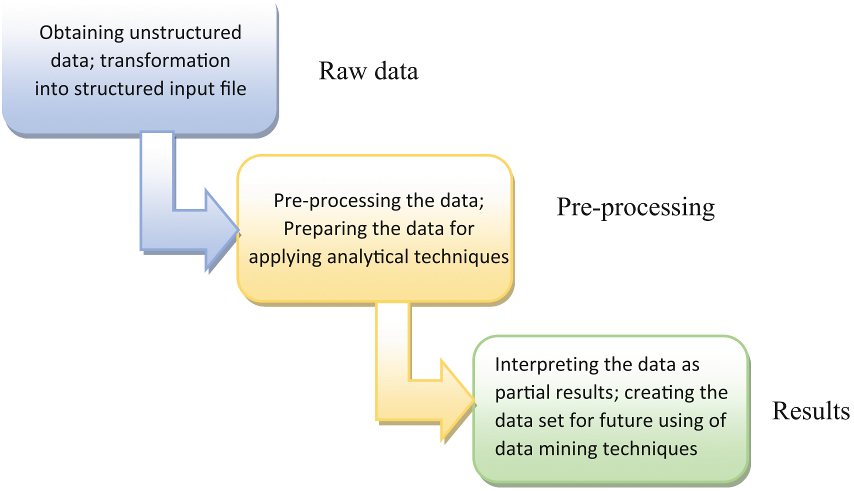

Obtaining unstructured data; transformation into structured input file
Raw data
Pre-processing the data; Preparing the data for applying analytical techniques
Pre-processing
Interpreting the data as partial results; creating the data set for future using of data mining techniques
Results

Fig. 1. The steps of knowledge discovery using text mining

(3) Apply Clustering (Descriptive Analytics) technique: The output from the "Process Documents from Data" operator consists of:
   (a) a word list,
   (b) a document vector.

The word list is not needed for clustering; however, the document vector is necessary. The output from the "Process Documents from Data" is obtained by pre-processing data interpreted as partial results for the company and consequently used as a basis data set for future analysis [6].

The CRISP-DM methodology was developed by the means of the effort of a consortium initially composed with Daimler-Chrysler, SPSS and NCR. CRISP-DM stands for CRoss-Industry Standard Process for Data Mining [13, 14]. It consists on a cycle that comprises six stages, captured on Fig. 2.

This initial phase, Business understanding, focuses on understanding the project objectives and requirements from a business perspective, then converting this knowledge into a data mining problem definition and a preliminary plan designed to achieve the objectives.

The Data understanding phase starts with an initial data collection and proceeds with activities in order to get familiar with the data, to identify data quality problems, to discover first insights into the data or to detect interesting subsets to form hypotheses for hidden information.

Subsequent Data preparation phase covers all activities to construct the final dataset from the initial raw data.

In the Modelling phase, various modelling techniques are selected and applied and their parameters are calibrated to optimal values.

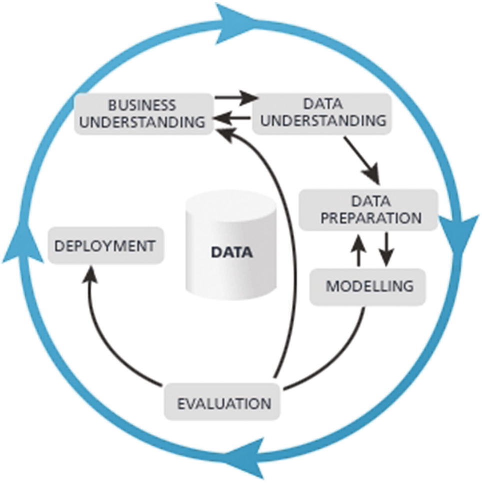


Fig. 2. Phases of CRISP-DM [10]

During the Evaluation stage, the obtained model (or models) is more thoroughly evaluated and the steps executed to construct the model are reviewed to be certain it properly achieves the business objectives.

Creation of the model is generally not the end of the project. The Deployment phase focuses on the organisation and presentation of knowledge and results in a way that the customer can further use.

The sequence of this six stages is not rigid. All these stages are duly organised, structured and defined, allowing that a project could be easily understood or revised [13].

Our proposal of knowledge discovery platform is based on the CRISP-DM methodology. Each phase of our approach corresponds with this methodology. The proposed paper shows only partial results from this process.

2.2 Get Data from Form to CSV Format

Acquired forms from car body works contain a set of data relating to breakdowns which occurred in this area. With regard to the availability of the data, we work only with big breakdowns and the duration of breakdowns is more than 30 min. All of attributes and values of breakdowns are stored in application forms. These forms are not suitable for the analysis and therefore are transformed using a program for extracting information from reports about major breakdowns in manufacturing.

Information on major breakdowns are stored in the similar structure for each breakdown, one report corresponds to one XLSX file. These reports are stored hierarchically in folders grouped by date of a breakdown.

For further processing of data, it is necessary to convert these reports to CSV format. Program for extracting this information is written in C# using Visual Studio 2015 as Windows form application. The program sequence is divided into the following sections:

- Select the topmost folder in the hierarchy.
- Select the file name and path where is extracted data stored.
- Find all relevant XLSX reports of major breakdown.
- Extract requested information from those reports.
- Get the information (parsing values in given cell range).
- Clean/transform the information.
- Save extracted information to a CSV file.

Since XLSX reports on major faults are stored in OpenXml format, library OpenXml was used to extract data from these reports [5]. In order to process a large number of reports, ongoing extraction and transformation of data is a parallel process. At the same time, 4 reports of breakdowns are processed.

2.3 RapidMiner

RapidMiner is currently one of the most used open source predictive analytics platforms for data analysis. It is accessible as a stand-alone application for information investigation and as a data mining engine for the integration into own products. Rapid miner provides an integrated environment for data mining and machine learning procedures, including [11]:

- extracting the data from different source systems; transforming the data and loading into a data warehouse (DW) or data repository other applications,
- data pre-processing and visualization,
- predictive analytics and statistical modelling, evaluation, and deployment.

What makes it even more powerful is that it provides learning schemes, models and algorithms from WEKA and R scripts [11].

Rapid miner provides a graphical user interface (GUI) to design and execute analytical workflows. Those workflows form a process, which consists of multiple Operators. A graphical user Interface (GUI) allows connecting the operators with each other in the process view. Each independent operator carries out one task within the process and forms the input to another operator in the workflow. The major function of a process is the analysis of the data which is retrieved at the beginning of the process [11].

Rapid miner offers large amount of different operators, which can be easily extended with existing extensions. There are packages for text processing, web mining, weka extensions, R scripting, series extension, python scripting, anomaly detection and more [7].

2.4 Text Mining Using RapidMiner

The RapidMiner Text Extension adds all operators necessary for statistical text analysis. Text from different data sources can be loaded and can be transformed by different filtering techniques, to analyse text data. The RapidMiner Text Extensions support several text formats including plain text, HTML, or PDF. It also provides standard filters for tokenization, stemming, stop word filtering, or n-gram generation. The Text

Processing package can be installed and updated through the Market RapidMiner menu item under the Help menu. The Text Mining extension uses a special class for handling documents, called the document class. This class stores the whole document in combination with additional meta information [8].

3 Design Process for Analysis

3.1 The Proposal of Text Mining Process

Figure 3 shows the proposed process, which contains operators and sub process called "Process Documentation from data". At the beginning of the process, there is "Read CSV" operator, which is used to read CSV files. The CSV files store text data in plain-text form, where all values of a record are in one line. Values of different attributes are separated by a column separator ",". Our CSV file contains 64 attributes in 358 records.

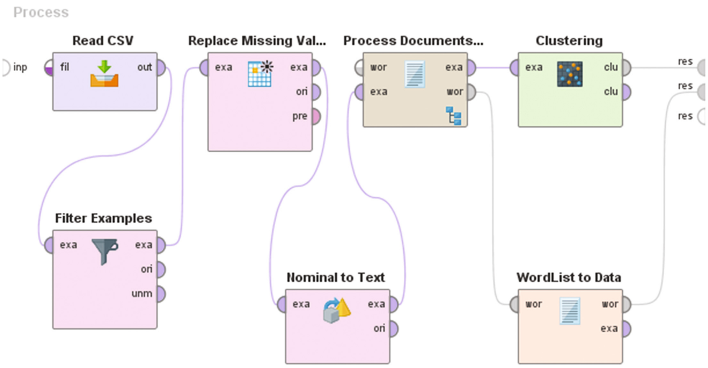


Fig. 3. The proposal of text mining process in RapidMiner

For our research, one of these attributes called "ResponsibleEmp" is important, as it contains information about the employee responsible for the breakdowns. Records in CSV format capture information about serious problems in production requiring repairs longer than 30 min.

"Filter example" operator removes the headers from the data of CSV files that remain in the data after joining multiple CSV, as well as several erroneous entries with a time correction less than 30 min. "Replace Missing Values" operator replaces missing values in examples of selected attributes "ResponsibleEmp" by the replenishment value of "missing". Next operator "Nominal to Text" changes the type of selected nominal attribute to the text and also ensures mapping of all values of attribute "ResponsibleEmp" to corresponding string values. After this process, the data is ready for text processing.

Subprocess operator called “Process Documents from data” you can see in Fig. 4. This operator generates word vectors from string attributes by using TF-IDF schema for creating corresponding vector in RapidMiner.

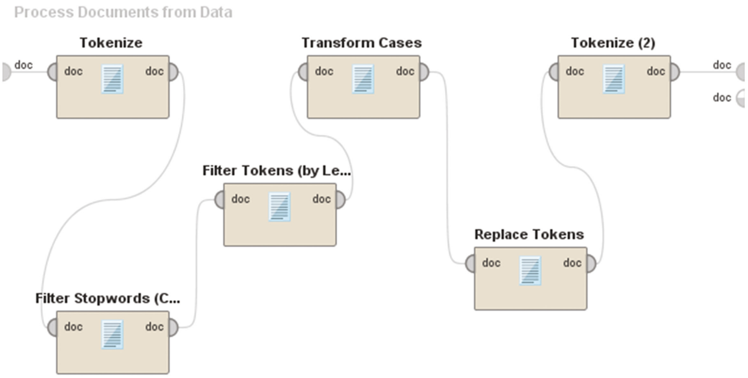


Fig. 4. Process documents from data

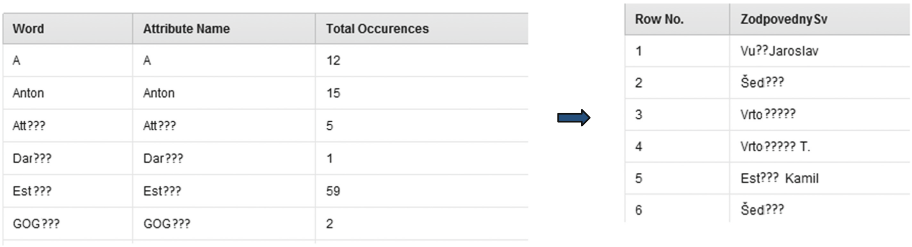
3.2 Tokenize

The first step in our process of text mining is using “Tokenize” operator. This operator splits the text of a document into a sequence of tokens. The simplest token is a character, although the simplest meaningful (to a human) token is a word. There are several options how to specify the splitting points (non-letters, specify characters, regular expression, linguistic sentences and linguistic tokens). The default setting is non-letters that will result in tokens consisting of one single word, what’s the most appropriate option before finally building the word vector [11]. In our paper, we do not split whole sentences, because in our data we can find only surnames or names. Some records involve non-standard character as “/, −” therefore we use tokenize with non-letters mode.

Figure 5, below, shows the effect of tokenizing on a single attributes called “ResponsibleEmp”. You can see that record with 2 or more values are split, punctuation mark characters are removed and tokens are separated. Some of values, mainly surname, are replace by ???, because these type of data are sensitive for company and must be anonymised.

3.3 Filter Stop Words

Next step in our pre-processing is used “Filter Stopword (Czech)” operator. This operator filters Czech stopwords from a document by removing every token which equals a stopword from the built-in stopword list. This operator filters common words as prepositions, conjunctions, adverbs and so on. It also reduces the unnecessary words and helps improve system performance [12]. Our records include only the names and surnames and do not contain any of those unnecessary words, therefore using this



Word	Attribute Name	Total Occurences
A	A	12
Anton	Anton	15
Att???	Att???	5
Dar???	Dar???	1
Est???	Est???	59
GOG???	GOG???	2

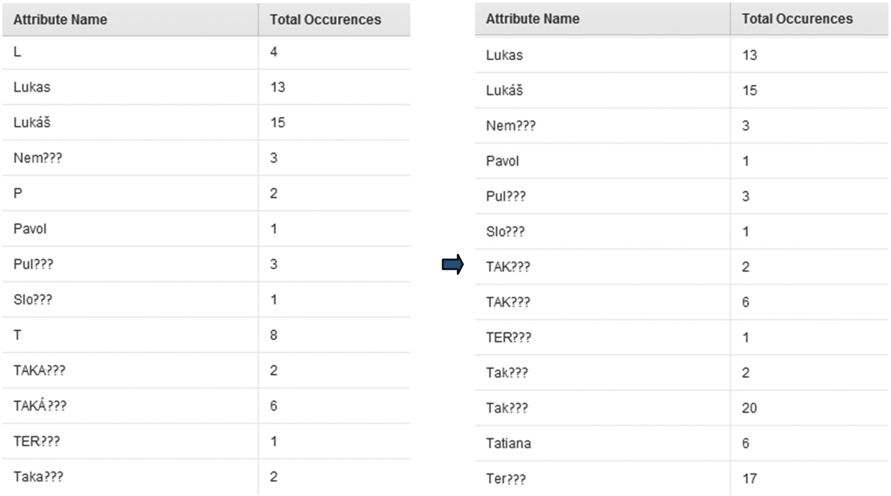
Row No.	ZodpovednySv
1	Vu??Jaroslav
2	Šed???
3	Vrto?????
4	Vrto????? T.
5	Est??? Kamil
6	Šed???

Fig. 5. Effect of tokenizing on a single attribute

operator is not necessary in our work. Nevertheless, this step is included in our proposed process, because "Filter stopword" operator is the second most commonly used in text processing (after tokenize) and for the future analysis, it can be very useful.

3.4 Filter Tokens (by Length)

The length of sequence of tokens can be taken into account. This operator filters tokens based on their length (i.e. the number of character they contain). Some records (name and surname) consist of academic degrees (in Slovak i.e. Ing., Bc., Mgr.), initials of names and this unnecessary information should be removed. Parameter "min chars" defines minimal number of characters that a token must contain to be considered and "max chars" describes the maximal number of characters allowed. Slovak academic degrees are usually not longer than 3 characters; therefore, the borders were set to 4–25. Some examples you can see in Fig. 6.



Attribute Name	Total Occurences
L	4
Lukas	13
Lukáš	15
Nem???	3
P	2
Pavol	1
Pul???	3
Slo???	1
T	8
TAKA???	2
TAKÁ???	6
TER???	1
Taka???	2

Attribute Name	Total Occurences
Lukas	13
Lukáš	15
Nem???	3
Pavol	1
Pul???	3
Slo???	1
TAK???	2
TAK???	6
TER???	1
Tak???	2
Tak???	20
Tatiana	6
Ter???	17

Fig. 6. The source and result of using filter tokens operator by length

3.5 Transform Case

In the process documents, we add operator “Transform Cases”. “Transform Cases” operator transforms all characters in a document to either lowercase or upper case. Transform case is necessary in order to avoid problems between similar words that differ in lowercase or uppercase. For example “GOLADA” is transformed to “golada”, expression “Ertrese” is converted to “ertrese”. Transform to lower case is very useful for next processing of text mining.

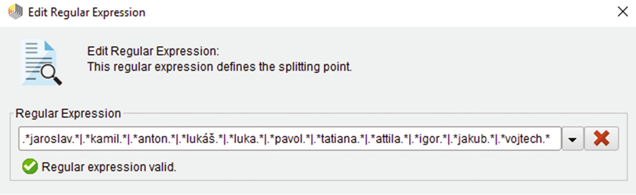
3.6 Replace Tokens

“Replace Tokens” another operator used, which allows replacing of substrings within each tokens. The user can specify arbitrary pattern-replacement-pairs in the replace_-dictionary parameter. The left column of the table specifies what should be replaced and the right column specifies the replacement.

In our case, Slovak language includes diacritical marks which should be removed and replaced by English characters for their easier use in the next step of document process, character “á” is replaced by “a”, “š” is replaces by “s” etc.

3.7 Tokenize (2)

The last operator applied is “Tokenize”, which is suitable for segregation of name and surname. As parameter “mode”, “regular expression” is selected. In a regular expression, a specific chars, as [a–z] any lowercase letter, .* multiple arbitrary character, etc. can be defined. Our regular expression removes names that are not carrying significant information. An example of a regular expression, you can see in Fig. 7.



Edit Regular Expression:
This regular expression defines the splitting point.

Regular Expression

```
.*jaroslav.*|.*kamil.*|.*anton.*|.*lukáš.*|.*luka.*|.*pavol.*|.*tatiana.*|.*attila.*|.*igor.*|.*jakub.*|.*vojtech.*
```

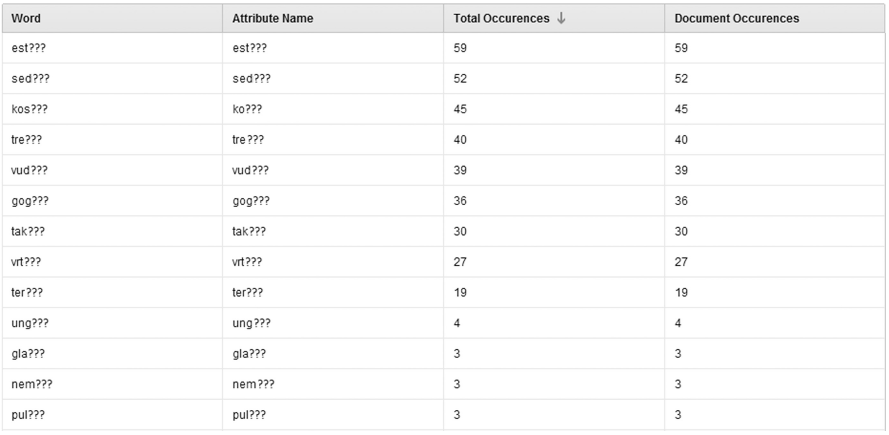
✓ Regular expression valid.

Fig. 7. Regular expression parameter

“Clustering” operator is prepared only for further research using advanced text mining techniques and “WordList to data” operator transform text data, as result of “Process Documents from Data”, into data format.

4 Evaluation and Results

The result of this research, which you can see in Fig. 8, is a list of responsible employees and a number of breakdowns they are responsible for.



Word	Attribute Name	Total Occurences ↓	Document Occurences
est???	est???	59	59
sed???	sed???	52	52
kos???	ko???	45	45
tre???	tre???	40	40
vud???	vud???	39	39
gog???	gog???	36	36
tak???	tak???	30	30
vrt???	vrt???	27	27
ter???	ter???	19	19
ung???	ung???	4	4
gla???	gla???	3	3
nem???	nem???	3	3
pul???	pul???	3	3

Fig. 8. List of responsible employee and count of breakdowns

We analysed data of a one-year period. On the basis of the results, company can design a system motivation and a method of the solution. Based on the acquired results, the given company can analyse the problem with a worker responsible for the largest number of breakdowns. The company can propose possible measures to improve the functioning of the production process.

The processed data provide a partial result for the company and also the data serves as a basic data set for the further research using advanced data mining techniques.

Acknowledgment. This publication is the result of implementation of the project VEGA 1/0673/15: “Knowledge discovery for hierarchical control of technological and production processes” supported by the VEGA.

This publication is the result of implementation of the project: “UNIVERSITY SCIENTIFIC PARK: CAMPUS MTF STU - CAMBO” (ITMS: 26220220179) supported by the Research & Development Operational Program funded by the EFRR.

References

1. Text mining, 18 March 2016. http://searchbusinessanalytics.techtarget.com/definition/text-mining
2. Paralič, J.: Dolovanie znalostí z textov. Košice, Equilibria (2010)
3. Al-Daihani, S.M., Abrahams, A.: A Text Mining Analysis of Academic Libraries’ Tweets. Elsevier, New York (2015)

4. Seo, W., Yoon, J., Park, H., Coh, B.-Y., Lee, J.-M., Kwon, O.-J.: Product Opportunity Identification Based on Internal Capabilities Using Text Mining and Association Rule Mining. Elsevier Inc. (2016)
5. Keyword clustering using web mining and text mining with RapidMiner, 22 March 2016. http://www.simafore.com/blog/bid/116340/Keyword-clustering-using-web-mining-and-text-mining-with-RapidMiner
6. Welcome to the open XML SDK 2.5 for office, 24 March 2016. <https://msdn.microsoft.com/en-us/library/office/bb448854.aspx>
7. Akthar, F., Hahne, C.: RapidMiner 5 operator reference (2014)
8. RapidMiner text mining extension, 01 April 2016. http://www.predictiveanalyticstoday.com/rapidminer-text-mining-extension/
9. Sheikhbahaei, M.R., Minaei, B.: Textual data mining applications in the automotive industry knowledge management and text classification (In Persian). In: The Eight Uran Data Mining Conference (2014)
10. Today's automotive markets must move beyond traditional strategies. https://rapidminer.com/industry/automotive/
11. Gupta, G., Malhotra, S.: Text documents tokenization for word frequency count using RapidMiner (Taking resume as an example). In: International Journal of Computer Applications (0975–8887), International Conference on Advancement in Engineering and Technology (2015)
12. Verma, T., Gaur, R.D.: Tokenization and filtering process in RapidMiner. Int. J. Appl. Inf. Syst. **7**(2), 16–18 (2014)
13. Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., Wirth, R.: CRISP-DM 1.0, step-by-step data mining guide. CRISP-DM Consortium (2000)
14. Michalik, P., Polačková, S., Zolotová, I.: Analysis of data from the monitoring environment to improve IT processes. In: Intelligent Engineering Systems, International Conference in Bratislava (2015)

Entropy Based Surface Quality Assessment of 3D Prints

Jarosław Fastowicz(✉) and Krzysztof Okarma

Department of Signal Processing and Multimedia Engineering,
Faculty of Electrical Engineering, West Pomeranian University of Technology,
Szczecin, 26 Kwietnia 10, 71-126 Szczecin, Poland
{jfastowicz,okarma}@zut.edu.pl

Abstract. In the paper the automatic method of visual quality assessment of surfaces of 3D prints is presented. The proposed approach is based on the use of entropy and may be applied for on-line inspection of 3D printing progress during the printing process. In case of observed decrease of the printed surface quality the emergency stop may be used allowing saving the filament, as well as possible correction of the printed object. The verification of the validity of the proposed method has been made using several prints made from different colors of the PLA filaments. Since the entropy of the image is related to the presence of structural information, the color to grayscale conversion of the test images has been applied in order to simplify further calculations. The analysis of the impact of the chosen color to grayscale conversion method on the obtained results is presented as well.

Keywords: 3D prints · Entropy · Image quality assessment · Image analysis

1 Introduction

One of the directions of development of image quality assessment methods is related to their applications in the world of 3D. Typically, such extension of metrics used for digital images is considered in view of the 3D graphics and several methods and specialized databases containing 3D images have been proposed during recent years. Most of them are intended for stereopairs [10,13,22] which can also be obtained as synthesized views [1] using Depth-Image-Based-Rendering (DIBR) techniques which are essential for the free-viewpoint television [18].

In order to verify the proposed image quality metrics useful for 3D images some databases containing subjective quality scores have been derived as well, e.g. IVP Anaglyph Image Database, IRCCyN/IVC NAMA3DS - COSPAD1 3D Video Quality Database, IRCCyN/IVC 3D Image Quality Database [2], LIVE 3D Image Quality Database [4,5] or MMSP 3D Video Quality Assessment Database [9], often together with the ideas of some new quality metrics. A short review of some of them can be found in the paper [15].

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_40

Nevertheless, according to our best knowledge, currently there are no available databases and metrics which are specific for the images of the 3D prints. Although the 3D printing technology becomes more and more popular and the price of available devices still decreases, there are almost no attempts to automatic quality assessment of the 3D prints.

Analyzing the use of machine vision for monitoring of the 3D printing process some interesting attempts can be noticed e.g. relatively old on-line defect detection for fused deposition of ceramics proposed by Fang [7] or method of monitoring the top surface of the print [6]. The first idea is based on the comparison of so called process signatures determined for the images captured by a camera and the reference images. The second approach utilizes the fuzzy model applied for the comparison of adjacent layers allowing identification of over- and underfilling during printing.

Another recent direction of research is the application of neural networks [21] for the quality assessment of 3D printed electronic products. Nevertheless, in many situations, the use of neural networks is considered as the "last choice" due to the necessity of learning process which is often hard to control and therefore the final results are not always predictable and satisfactory. Such a system applied to a 3D inkjet printer is based mainly on the resistivity measurement together with comparison of shape and geometrical properties so its further combination with more advanced machine vision approach seems to be promising direction of research. Description of some other similar applications can be found in the Szkilnyk's [20] and Chauhan's [3] papers.

The most promising paper related to automatic correction of detected errors in desktop 3D printers has been published by Straub [19]. Nevertheless, the system presented in the paper, based on Raspberry Pi modules and five cameras, has been implemented only at a very initial stage and requires many interruptions of the printing process. Although it allows the detection of lack of filament (leading to so called "dry printing"), it is very sensitive to distortions caused by camera motion, dynamic lighting conditions etc. Due to the requirement of precise calibration, it is rather hard for practical implementation, especially for home use 3D printers.

In our earlier papers some attempts to the automatic quality assessment of 3D prints based on texture analysis and local similarity have been presented. The idea based on the analysis of the Gray Level Co-occurrence Matrix (GLCM) allows obtaining the no-reference metrics [8,14] but unfortunately is quite slow due to high amount of computations. Nevertheless, the application of Feature Similarity [16] and Structural Similarity based metrics [17] allows relatively fast and usually proper classification of 3D prints into lower and higher quality groups mainly for scanned images of 3D printed surfaces. Since the scanned images are characterized by more uniform lighting conditions, such an approach has also been used in this paper in order to verify the usefulness of proposed entropy based metric.

Considering the fact that most image quality metrics, including the Structural Similarity and Feature Similarity applied in both papers, can be computed for grayscale images, the necessity of color to grayscale conversion influences the obtained results as well. Similar verification for the entropy based metric is presented in further part of the paper as well.

2 The Idea of Entropy Based Metric

The idea of the application of entropy for the quality assessment of 3D prints originates from the assumption that the ideal surface of the 3D print observed by the side view camera can be characterized by regular linear patterns representing consecutive layers of the filament. In the presence of distortions this structure becomes corrupted and the observed pattern is more complicated. In such case the amount of information visible on the image plane increases and therefore the entropy of such image should be higher. For highly distorted surfaces of 3D prints the values of entropy should be much higher allowing the classification of the 3D prints.

From theoretical point of view image entropy is the statistical measure of randomness defined as:

$$E = -\sum_{i=1}^{N} p_i \cdot log_2(p_i) \tag{1}$$

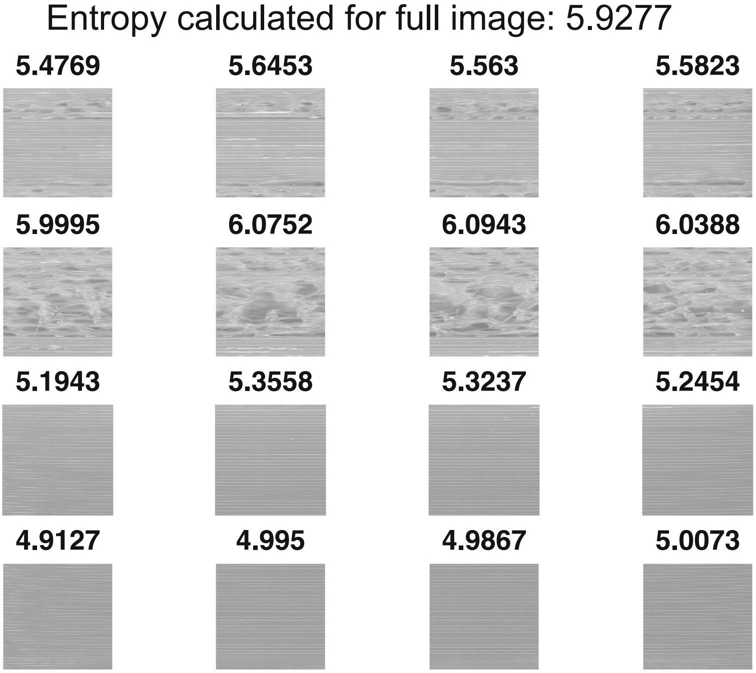


Fig. 1. Local entropy values for exemplary orange sample converted to greyscale with varying quality.

where the vector p contains the histogram counts calculated assuming N=256 bins for grayscale images.

Since the entropy values can be considered as the measure of amount of distortions, one may expect its high correlation with the perceived quality of the 3D printed surfaces. The additional advantage of the proposed use of entropy as the quality measure is the possibility of its local calculation for the specified fragment of the printed surface. Such an approach allows its utilization for the on-line monitoring of the 3D print quality during the printing process. In the case of increase of the local entropy in comparison to the values calculated for previously printed bottom parts of the object, some problems may be detected and the specified action may be taken. The illustration of the local entropy values for an exemplary sample of the 3D print with varying quality is presented in Fig. 1 where high entropy values are characteristic for the samples containing noticeable distortions.

3 Details of Experiments

Verification of the usefulness of the proposed approach for the automatic visual quality assessment of 3D prints has been made using several flat sample prints obtained using two 3D printers based on Fused Deposition Modeling (FDM) technology, namely RepRap Pro Ormerod 2 and Prusa i3. Several different colors sample plates made from bio-degradable polylactic acid (PLA) filaments have been obtained with forced local decrease of print quality. The lower quality samples have been obtained by hanging the temperature and modifications of speed of the filament's delivery speed. Some of them are presented in Fig. 2.

Such plates have been scanned from both sides with 1200 dpi resolution and additionally divided into parts in order to compute the local quality indicators as shown in Fig. 1. Captured images have been converted into grayscale using four commonly used methods based on popular color models and the luminance has been calculated as the maximum of RGB channels, the average of the RGB channels and as the weighted average according to the ITU recommendations:

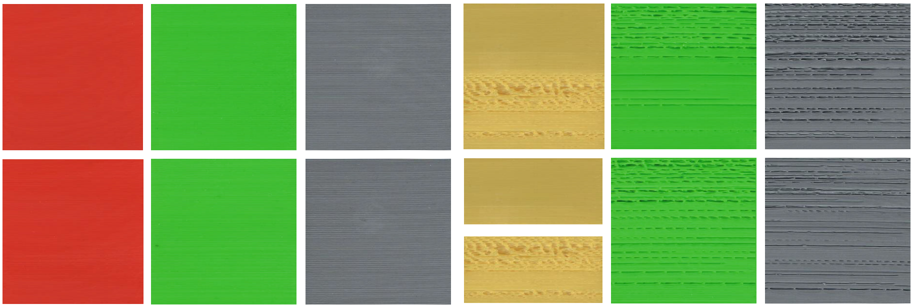


Fig. 2. Exemplary scans of the high and low quality 3D prints used in experiments.

BT.601-7 [11] used e.g. in MATLAB *rgb2gray* function and BT.709-6 [12] used mainly in HDTV.

For all scanned samples the global entropy values have been calculated as well as the local values obtained dividing the images for 4 and 16 parts. All the scans have been made in two ways scanning the samples perpendicularly to the filament's layers and in parallel to them. The results of the calculations assuming different color to grayscale conversion methods are illustrated in Figs. 3, 4, 5 and 6. The samples have been numbered such that the numbers 1–18 indicate the samples scanned perpendicularly whereas the numbers 19–36 denote the parallel scans. The colors of plots are typically related to the colors of filaments (however white filament is marked as blue and the silver one using black symbols for better visualization). The circles denote high quality prints whereas the crosses stand for low quality samples. The orange samples have varying quality and therefore big differences in their local entropy can be observed after the division

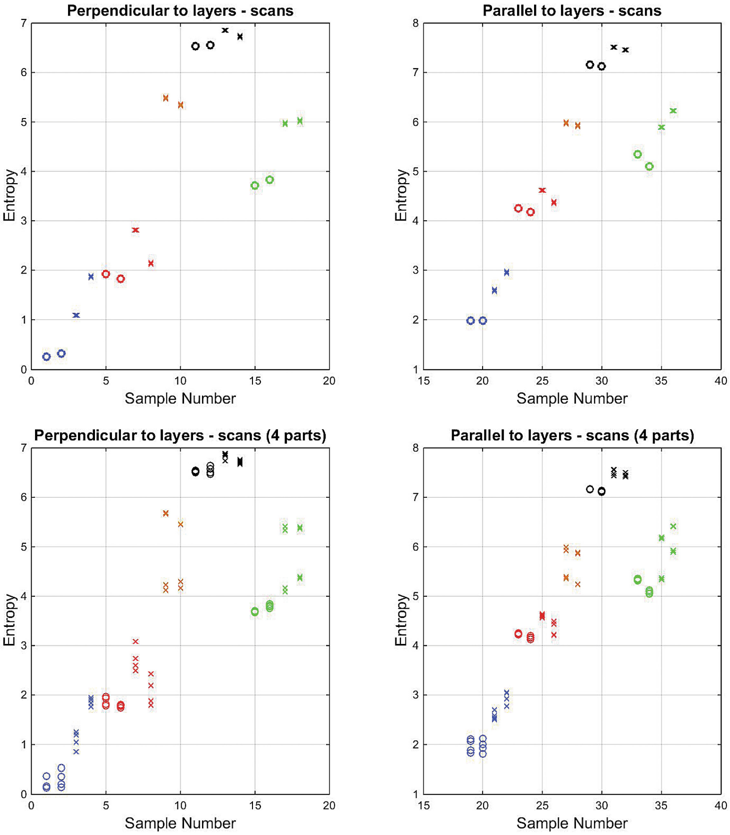


Fig. 3. Entropy values for different colors of samples and their local values obtained for division into 4 parts (o - high quality prints, x - low quality prints) using the grayscale conversion according to recommendation ITU BT.601-7.

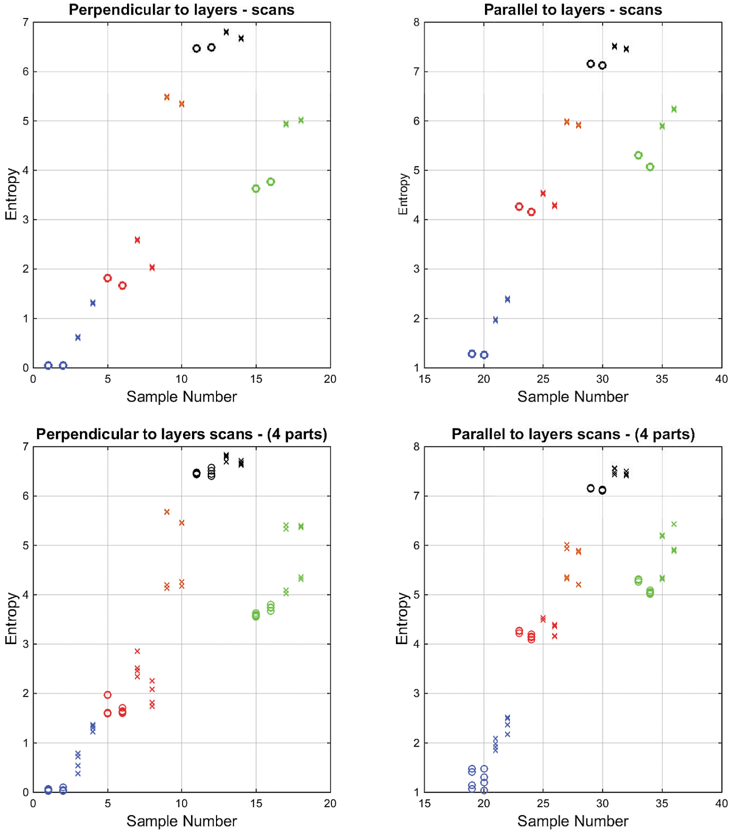


Fig. 4. Entropy values for different colors of samples and their local values obtained for division into 4 parts (o - high quality prints, x - low quality prints) using the grayscale conversion according to recommendation ITU BT.709-6.

of the scanned images into 4 or 16 regions. A similar effect, caused by the presence of local distortions not covering the whole surface, can also be observed for the red samples although the differences are not as high.

Results obtained for perpendicular and parallel scans as well as for different color to grayscale conversion methods differ noticeably but lead to similar conclusions. Nevertheless for perpendicular scans the differences between the entropy for high and low quality samples are higher allowing easier classification of samples.

Presented approach makes possible to identify the decrease of the 3D printed surface quality during the printing process assuming the possibility of mounting the camera such that the side view of the printed object can be observed. Since the color of the filament as well as color to grayscale conversion method have a great impact on the obtained absolute entropy values, the proposed method can be successfully implemented assuming the availability of the reference high

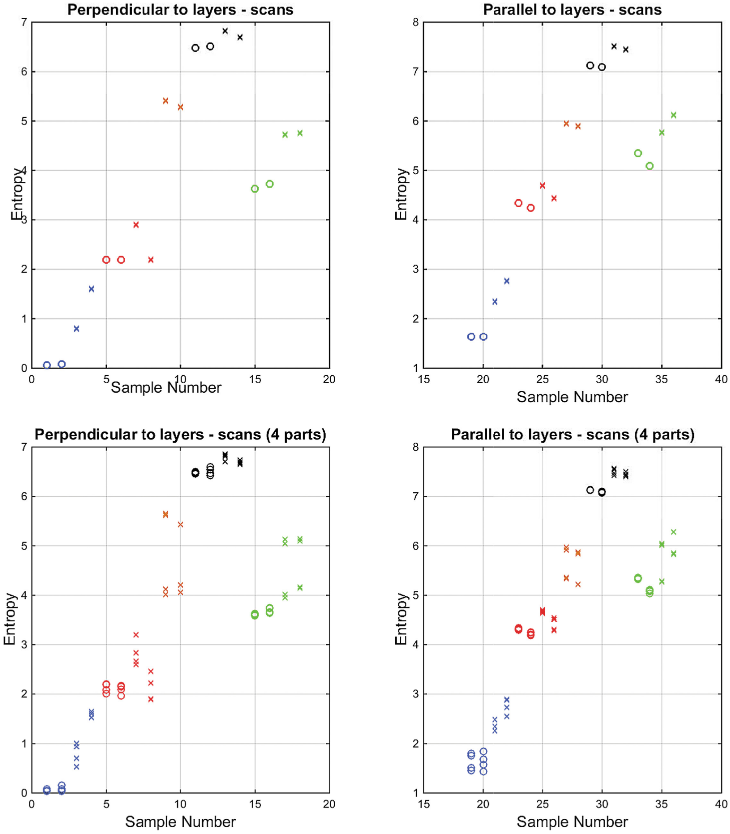


Fig. 5. Entropy values for different colors of samples and their local values obtained for division into 4 parts (o - high quality prints, x - low quality prints) using the grayscale conversion as the average of the RGB channels.

quality fragment of the printed surface using the same color of the filament. In many cases it can be just the bottom part of the printed sample.

Nevertheless a proper detection of the decreased quality or using the entropy values for the quality assessment in the continuous quality scale would require the knowledge of the reference entropy values for the perfect quality 3D print assuming the specified color of the filament and color to grayscale conversion method. At the current stage of research it seems to be one of the most relevant limitations of the proposed approach. Therefore one of the directions of our further experiments will be the implementation of the no-reference entropy based method for the quality assessment of the 3D prints where the expected entropy value should be predicted on the basis of the filament's color.

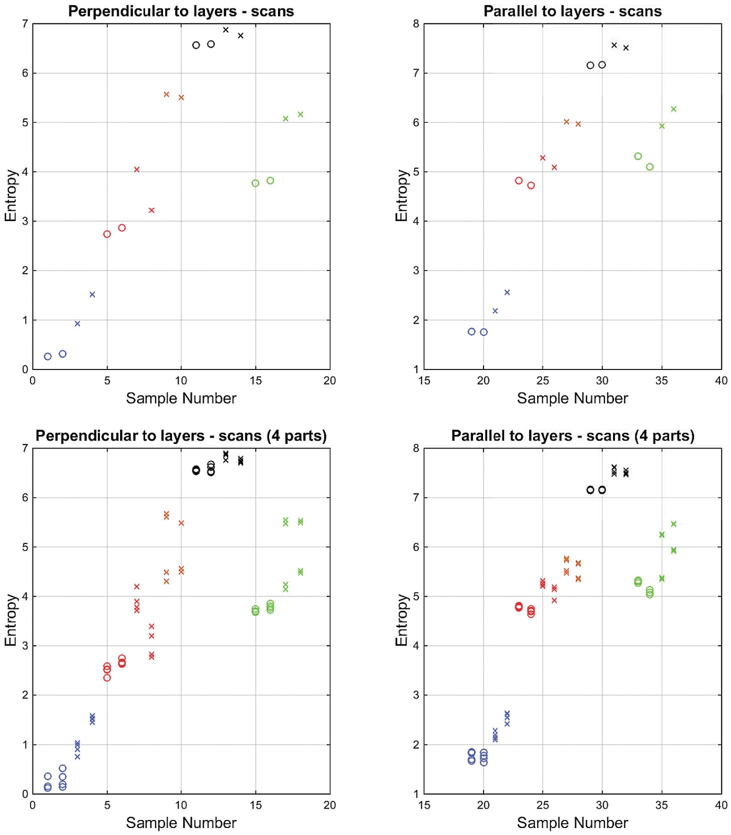


Fig. 6. Entropy values for different colors of samples and their local values obtained for division into 4 parts (o - high quality prints, x - low quality prints) using the grayscale conversion as the maximum of the RGB channels.

4 Concluding Remarks

Automatic visual quality assessment of the 3D prints still remains one of the challenges of image analysis especially in view of the most desired no-reference methods. Development of the universal method suitable for all types of 3D prints seems to be doubtful but some methods e.g. based on texture analysis can be successfully applied utilizing the specificity of the surface patterns obtained using the most popular PLA based devices.

The application of the entropy for such purposes leads to very promising results allowing not only the classification of the 3D prints into low and high quality prints but also the identification of the distorted fragments of the surface. Computing the local entropy values it is possible to obtain the quality map of the surface and, in the long run, express the quality of the surface in a continuous quality scale. In such solution the quality metric would give the information not only about the presence of distortions but also about their amount.

Although the presented approach have its limitations e.g. does not utilize shape information, it can be combined with some other methods. Such a combination, particularly with the texture analysis methods discussed in our earlier papers [8,14,16], will be one of the most important challenges of our further research.

References

1. Battisti, F., Bosc, E., Carli, M., Callet, P.L., Perugia, S.: Objective image quality assessment of 3D synthesized views. Sig. Process. Image Commun. **30**, 78–88 (2015)
2. Benoit, A., Le Callet, P., Campisi, P., Cousseau, R.: Quality assessment of stereoscopic images. EURASIP J. Image Video Process. **2008**(1), 659024 (2008)
3. Chauhan, V., Surgenor, B.: A comparative study of machine vision based methods for fault detection in an automated assembly machine. Procedia Manufact. **1**, 416–428 (2015)
4. Chen, M.J., Cormack, L.K., Bovik, A.C.: No-reference quality assessment of natural stereopairs. IEEE Trans. Image Process. **22**(9), 3379–3391 (2013)
5. Chen, M.J., Su, C.C., Kwon, D.K., Cormack, L.K., Bovik, A.C.: Full-reference quality assessment of stereopairs accounting for rivalry. Sig. Process. Image Commun. **28**(9), 1143–1155 (2013)
6. Cheng, Y., Jafari, M.A.: Vision-based online process control in manufacturing applications. IEEE Trans. Autom. Sci. Eng. **5**(1), 140–153 (2008)
7. Fang, T., Jafari, M.A., Bakhadyrov, I., Safari, A., Danforth, S., Langrana, N.: Online defect detection in layered manufacturing using process signature. In: Proceedings of IEEE International Conference on Systems, Man and Cybernetics, vol. 5, pp. 4373–4378, San Diego, California, USA (1998)
8. Fastowicz, J., Okarma, K.: Texture based quality assessment of 3D prints for different lighting conditions. In: Chmielewski, L.J., Datta, A., Kozera, R., Wojciechowski, K. (eds.) ICCVG 2016. LNCS, vol. 9972, pp. 17–28. Springer, Cham (2016). doi:10.1007/978-3-319-46418-3_2
9. Goldmann, L., Simone, F.D., Ebrahimi, T.: A comprehensive database and subjective evaluation methodology for quality of experience in stereoscopic video. In: 3D Image Processing (3DIP) and Applications, vol. 7526 in Proceedings of SPIE (2010)
10. Guo, J., Vidal, V., Cheng, I., Basu, A., Baskurt, A., Lavoue, G.: Subjective and objective visual quality assessment of textured 3D meshes. ACM Trans. Appl. Percept. **14**(2), 11:1–11:20 (2016)
11. International Telecommunication Union: Recommendation ITU-R BT.601-7 - Studio encoding parameters of digital television for standard 4:3 and wide-screen 16:9 aspect ratios (2011)
12. International Telecommunication Union: Recommendation ITU-R BT.709-6 - Parameter values for the HDTV standards for production and international programme exchange (2015)
13. Lin, Y., Wu, J.: Quality assessment of stereoscopic 3D image compression by binocular integration behaviors. IEEE Trans. Image Process. **23**(4), 1527–1542 (2014)

14. Okarma, K., Fastowicz, J.: No-reference quality assessment of 3D prints based on the GLCM analysis. In: 2016 21st International Conference on Methods and Models in Automation and Robotics (MMAR), pp. 788–793 (2016)
15. Okarma, K.: On the usefulness of combined metrics for 3D image quality assessment. In: Choraś, R.S. (ed.) Image Processing and Communications Challenges 6. AISC, vol. 313, pp. 137–144. Springer, Heidelberg (2015). doi:10.1007/978-3-319-10662-5_17
16. Okarma, K., Fastowicz, J.: Quality assessment of 3D prints based on feature similarity metrics. In: Choraś, R.S. (ed.) Image Processing and Communications Challenges 8, pp. 104–111. Springer, Heidelberg (2017). doi:10.1007/978-3-319-47274-4_12
17. Okarma, K., Fastowicz, J., Tecław, M.: Application of structural similarity based metrics for quality assessment of 3D prints. In: Chmielewski, L.J., Datta, A., Kozera, R., Wojciechowski, K. (eds.) ICCVG 2016. LNCS, vol. 9972, pp. 244–252. Springer, Cham (2016). doi:10.1007/978-3-319-46418-3_22
18. Starch, J., Kilner, J., Hilton, A.: Objective quality assessment in free-viewpoint video production. In: Proceedings of the 3DTV Conference: The True Vision - Capture. Transmission and Display of 3D Video, pp. 225–228, Istanbul, Turkey (2008)
19. Straub, J.: Initial work on the characterization of additive manufacturing (3D printing) using software image analysis. Machines **3**(2), 55–71 (2015)
20. Szkilnyk, G., Hughes, K., Surgenor, B.: Vision based fault detection of automated assembly equipment. In: Proceedings of ASME/IEEE International Conference on Mechatronic and Embedded Systems and Applications, Parts A and B, vol. 3, pp. 691–697, Washington, DC, USA (2011)
21. Tourloukis, G., Stoyanov, S., Tilford, T., Bailey, C.: Data driven approach to quality assessment of 3D printed electronic products. In: Proceedings of 38th International Spring Seminar on Electronics Technology (ISSE), pp. 300–305, Eger, Hungary, May 2015
22. Yang, J., Hou, C., Zhou, Y., Zhang, Z., Guo, J.: Objective quality assessment method of stereo images. In: Proceedings of the 3DTV Conference: The True Vision - Capture. Transmission and Display of 3D Video, pp. 1–4, Potsdam, Germany (2009)

Dim Line Tracking Using Deep Learning for Autonomous Line Following Robot

Grzegorz Matczak(✉) and Przemysław Mazurek

Department of Signal Processing and Multimedia Engineering,
West–Pomeranian University of Technology, 26. Kwietnia 10 Street,
71126 Szczecin, Poland
grzegorz.matczak@gmail.com, przemyslaw.mazurek@zut.edu.pl

Abstract. The proposed approach improves preprocessing of image data for the line following robot. The tracking algorithm uses Track–Before–Detect algorithm using Viterbi algorithm. Proposed technique uses deep learning for the estimation of the line and background area. The segmentation improves detection of weak line on the image disturbed by numerous additive patterns and Gaussian noise.

Keywords: Line following robot · Track–Before–Detect · Viterbi algorithms · Deep learning · Convolutional Neural Networks

1 Introduction

Line following robots are applied in numerous applications. There are two main types of lines: high contrast and low contrast. The first group of lines is man–made line for the determination of robots movement. Such lines are available in factories and line following robots limit movement trajectory to defined areas. The second group of lines is natural, that exist in interior or exterior environment. Such lines could be real lines, like horizontal lines on roads or edge between two areas, e.g. area with different plants, area before/after harvesting [1]. There are numerous applications, e.g. computer assisted harvesting, trash compacting, snow or sand removal from runways [14]. Recent advantages in UAVs extend application to the power line inspections for example [4]. Line tracking in different images, like medical images are considered in [15].

Existing lines could be low contrast with noise and numerous disturbances, including influence of light condition so conventional tracking systems are not feasible. The conventional tracking system uses detection, tracking and assignment for improving tracing quality in real (noised) environment. There are different sensing systems in line following robots, and optical systems are very important but sensitive to numerous disturbances. Two simple sensors could be used for line tracking, but linear optical sensors or cameras are more useful, because larger area is observed. Increased area of observation where the possible line exists, allows the improvement of the tracking.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_41

In the case of real dim line, the conventional tracking system fails, due to high ratio of false observations or missed observations of line pixels. The problem is the detection algorithm that convert grayscale image to binary using threshold algorithm. The application of fixed threshold is very limited, but even an adaptive threshold algorithm fails if SNR (Signal–to–Noise Ratio) is low. The solution is possible using only an alternative processing scheme - Track–Before–Detect (TBD) [16]. This approach uses processing (tracking) without prior detection (thresholding). All possible trajectories are analyzed, even if no object is in the range, so computation cost is significant.

Tracking algorithms could be suitable for objects that are points or blobs on image, but the line tracking is possible using single image frame. The sequences of frames could be combined together, for visual servoing, because line observations are necessary for the computation of error between line and current trajectory of robot, for further correction of robot movement.

The raw images could be delivered to conventional or TBD algorithm. Image enhancement using image processing algorithm could be applied for improving performance [10]. It is also necessary if direct raw data cannot be delivered to algorithm. TBD algorithms expect positive signal from object and zero mean noise related to background. The background estimation or background removal is important for enhancement of image. The application of detection of specific features in the image could be applied for improving SNR also. Linear filters, nonlinear (e.g. morphological) filters remove spatial noise, that is important technique.

Advanced algorithm, using feature detection related to the object or background features could be also applied. This technique is important for non-trivial point objects or blob objects. The object could be extended object that contain some spatial features and occupy tens or hundreds of pixels. Pattern recognition techniques are very usefully for improving SNR.

1.1 Related Works

Numerous tracking algorithms, related to civil or military applications, are proposed [2]. The line detection could be processed using Hough or Radon Transforms [3], but both mentioned algorithms are limited to straight lines. Viterbi algorithm [5,17] is applied for line tracking directly [9]. The conventional tracking filters like the Benedict–Bordner, Kalman, or EKF could be also applied if the detection algorithm is applied [2]. TBD algorithms could be applied for line following robots using Viterbi algorithm directly or after preprocessing [11,13]. ConvNets (Convolutional Neural Networks) are applied in numerous tasks, including improvement of edge detection system [18].

1.2 Content and Contribution of the Paper

The application of ConvNet for the line tracking is proposed in this paper. Numerical simulations are considered for assumed model of image. The generator of synthetic images is applied for the testing of performance.

Proposed architecture of line following system is considered in Sect. 2. Exemplary results for this system are presented in Sect. 3. Discussion is provided in Sect. 4 and the final conclusions are in Sect. 5.

2 Application of Convolutional Neural Networks for Improvement Tracking

Artificial Neural Networks (ANN) are applied for numerous data processing tasks, like pattern recognition, image processing or image compression. There are a few disadvantages of ANN: long learning time and lack of knowledge about obtained processing technique. Both problems are addressed in ConvNets [7] where limited number of weights is trained, because full connection between layers is generally not used. The input image is processed using set of kernels with sliding window approach. The learning process allows fifing of weight of kernels to the specific features that are available on images. The reduction of size of intermediate images (results of kernel processing) is obtained by specific layers. The combination of convolutional layers, reduction layers and other gives deep neural network [8], because many layers (tens of them) are used. Conventional NN uses a few layers only. The obtained hierarchy is similar to the concept of Neo–Cognitron, where the beginning layers are detectors of basic features, the intermediate layers are related to detection of higher order of features and the ending layers are main classifier.

The proposed system consists of ConvNet and TBD subsystem (Fig. 1)

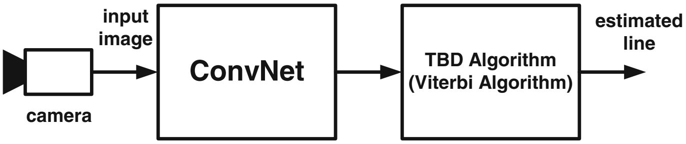


Fig. 1. Line following robot tracking system

The main task of ConvNet is the improving of detection of line and suppression of other features. This part of system is learned using training pair (input and output images). Both images are obtained from image generator. The learning process requires numerous presentations of training pair, so the validation of proposed technique image generator is applied. The real world images could be also applied if the proper augmentation is used for increasing of number of training pairs. The problem with real images is related to the availability of output image that is necessary for supervised learning. Lines should be defined by human and creation of such database is time consuming process.

Proposed structure of ConvNet is depicted in Fig. 2:

The second part of proposed system is the Viterbi algorithm. This algorithm is not considered in this paper. Explanation of this algorithm in context of line tracking and preprocessing possibilities are available in another papers [11,13].

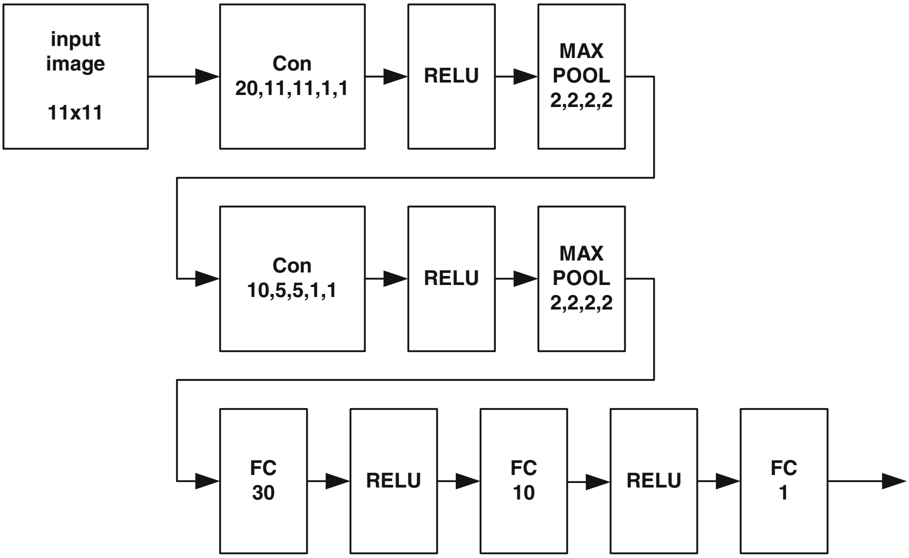


Fig. 2. Proposed ConvNet

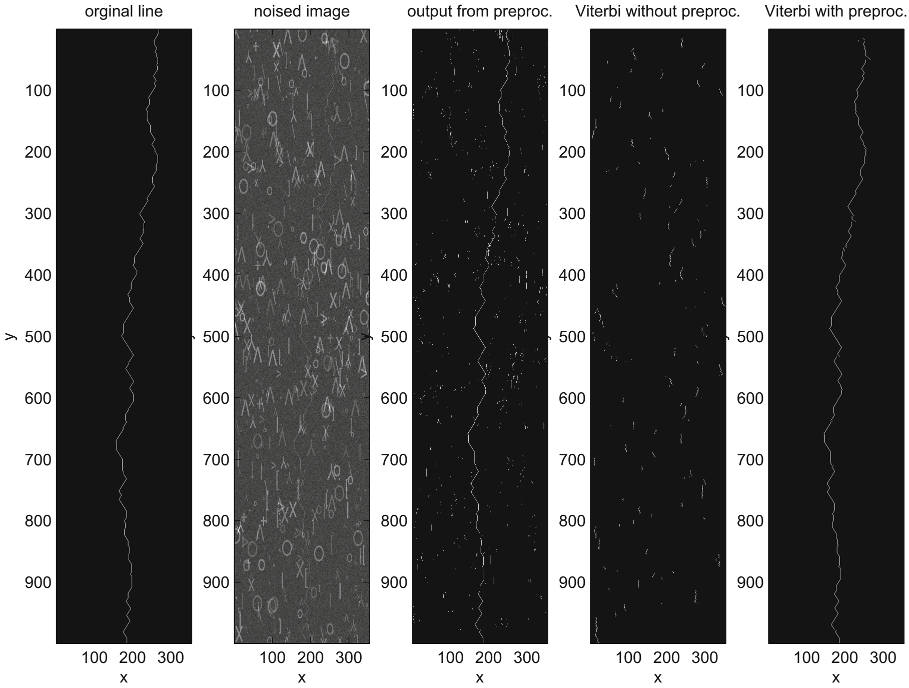


Fig. 3. Comparison of results: depth of analysis 16 rows, std. dev. 0.2

3 Experiments and Results

The input data are processed using sliding window during testing phase. The learning process uses random positions of window. The learning process uses mini–batch processing using 15000 images at one GPU processing cycle. This large set, and requires 12 GB of GPU memory.

The learning time is about a two hour using single learning image with medium level of disturbances and noise. All images with different levels of disturbances and noise where tested directly, without additional learning. The system for training of ConvNet uses Intel i7 2.66 GHz processor with 6 GB RAM. ConvNet processing uses NVidia GeForce Titan X (Maxwell) GPU (12 GB RAM). Software for training is based on dlib–19.2 [6] (C++11 code) and Nvidia cuDNN v7.5, with custom data augmentation and manual fitting to maximization of performance (GPU RAM, PCI Express and CPU utilization).

The motion model of line is defined by random process with three motion vectors: $-1, 0, +1$. The line length is also controlled by random number generator. The assumed width of line is single pixel. The availability of true position of the line allows testing algorithm. Such image is also applied in training pair.

The influence of additive Gaussian noise is tested together with larger disturbances - random patterns. Selected letters and signs patterns are used in

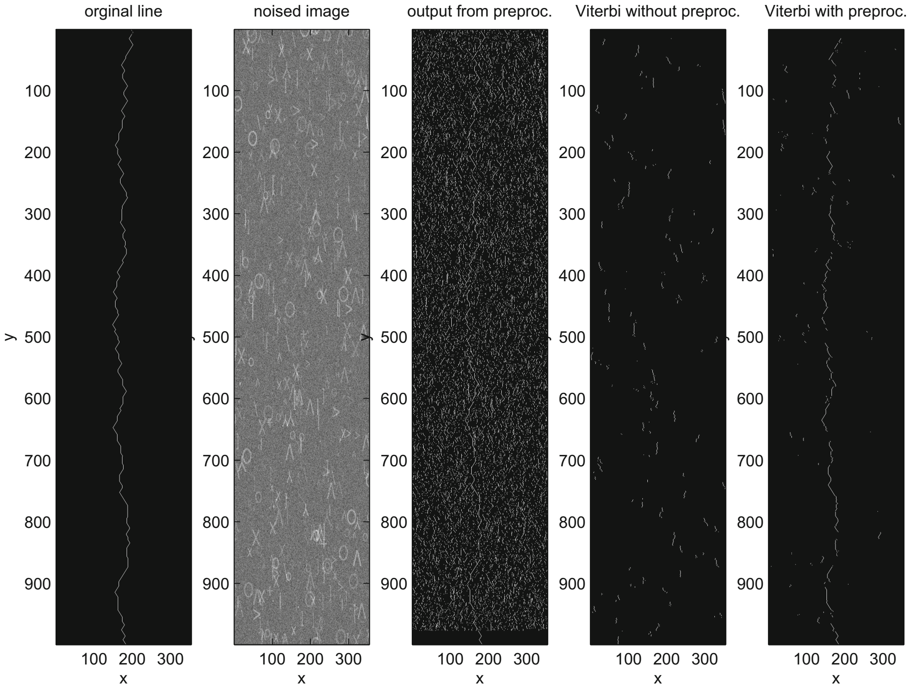


Fig. 4. Comparison of results: depth of analysis 16 rows, std. dev. 0.5

this test, so different disturbances for Viterbi algorithm like horizontal splitting, merging, parallel line, horizontally wide and strong signals are available.

Viterbi algorithm process selected number of rows for the estimation of line. This depth of analysis influences the results: shallow analysis is not robust, deeper analysis gives better results typically, but is processing time consuming.

The line is not visible to the human due to noise and disturbances, but the visibility could be improved using ConvNet (Figs. 3, 4, 5, and 6). The original line, with disturbances and noises are depicted. The output of Viterbi algorithm without preprocessing and with preprocessing are provided for different depths of analysis.

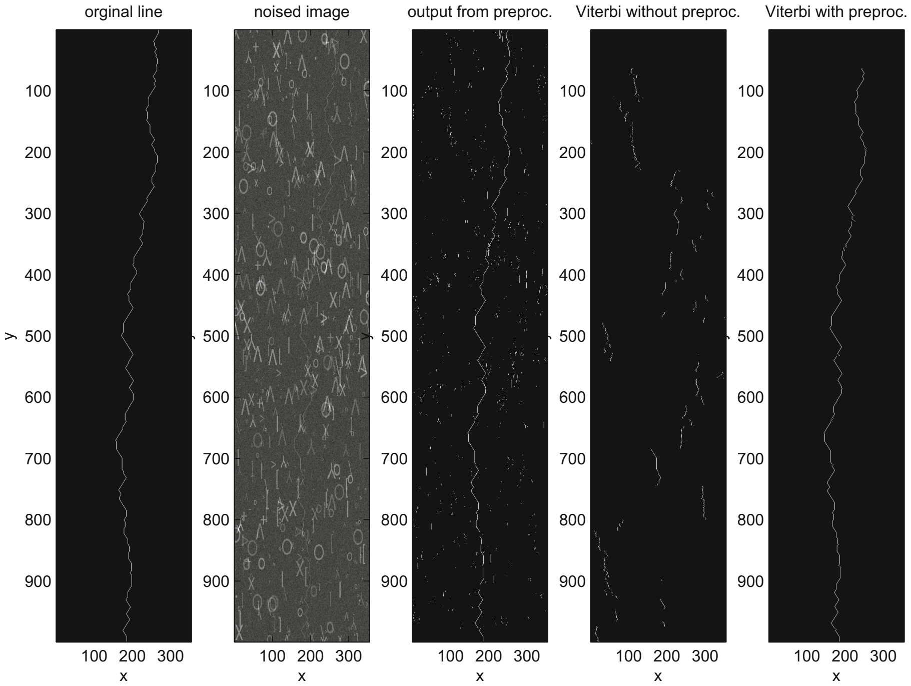


Fig. 5. Comparison of results: depth of analysis 64 rows, std. dev. 0.2

The mean of cumulative error (distance between true and estimated position) for 1000 rows are depicted in Fig. 7. The mean error for overall image is about 10 pixels for preprocessing and 100 for case without preprocessing. The errors are large for some rows, but are zero value for others.

It is well visible for magnified view (Fig. 8) and the increasing of depth of analysis reduced problem of discontinuity of line. The lack of discontinuity is the internal behavior of Viterbi algorithm that uses a selected number of rows (defined by depth of analysis), but does not use previous results.

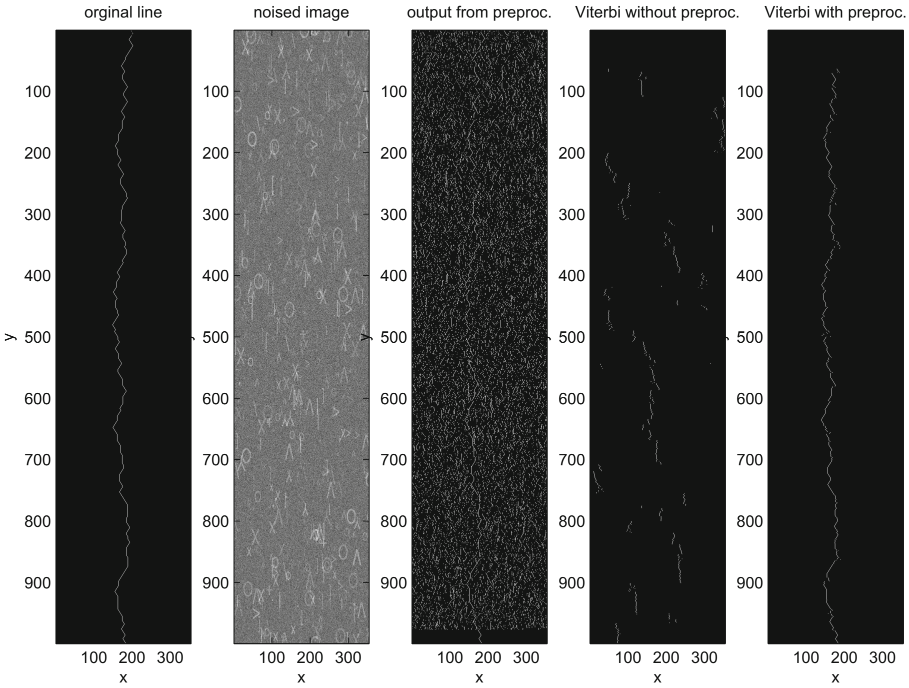


Fig. 6. Comparison of results: depth of analysis 64 rows, std. dev. 0.5

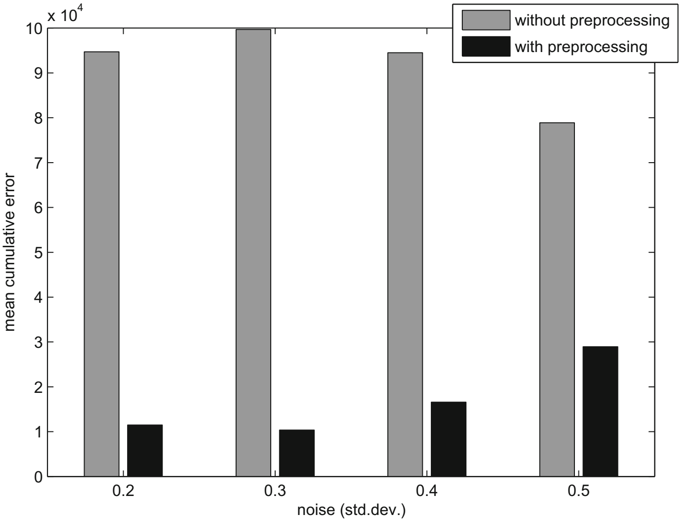
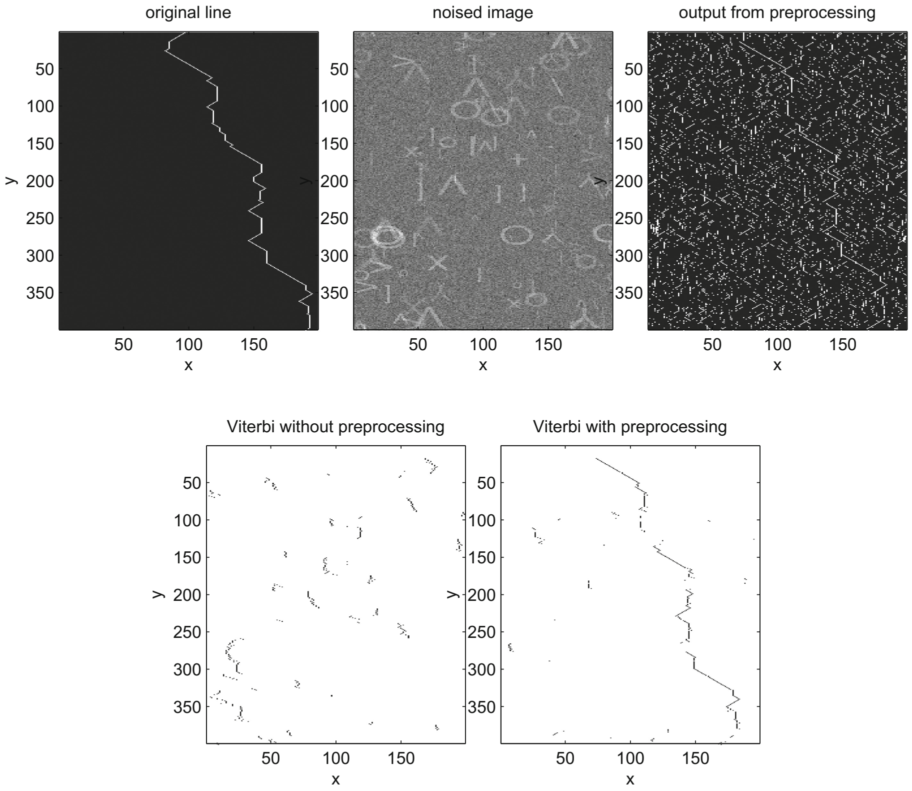


Fig. 7. Comparison of results: depth of analysis 16 rows


original line

noised image

output from preprocessing

Viterbi without preprocessing

Viterbi with preprocessing

Fig. 8. Comparison of results - details of processing

4 Discussion

Presented examples show problem of Viterbi algorithm. Low level of disturbances and noise allow partial estimation of line (Fig. 3). Obtained result is useless for depth 16, some parts of the line are estimated for depth 64 (Fig. 5). Large noise reduces possibility of the line estimation.

The proposed solution using ConvNet preprocessing gives binary output and white line is visible. There are a lot of false detection, so salt noise is observed. Such false detections are related to all image areas, including line neighborhood pixels. There are some problems if depth is shallow (16 rows) and noise is high.

Viterbi algorithm supports TBD processing, so raw image data could be processed directly. Additive noise disturbs processing, but due to the detection of line using multiple image rows it is possible to estimate correct position of line in each row. Additive disturbance adds essential problems, because accumulative approach in TBD algorithm is the source of high values preference. Short false line with high values should be suppressed

The application of ConvNet allows the segmentation of image, pixel by pixel and binary output is delivered to Viterbi algorithm, that is acceptable. High disturbances and noise level reduce performance of overall system, but it is not possible to process image using TBD approach directly. Conventional tracking algorithms are not suitable for such task.

Further improvements of line estimation are possible using line filtering algorithm.

5 Conclusions and Further Work

Proposed approach uses learning process for preprocessing part of system. Additional knowledge added to system improves greatly overall performance, that is shown in the paper. TBD algorithm could be improved using learning techniques in preprocessing part of the tracking system. The learning is not related only to the background, but for line and both of them together also.

The implementation of the system for real robot will be addressed in further work. Real–time Viterbi algorithm implementations are available and ConvNet real–time processing is possible also.

Combination of multiple TBD algorithms also improves tracking [12], so further improvement of performance is possible.

Acknowledgments. This work is supported by the UE EFRR ZPORR project Z/2.32/I/1.3.1/267/05 "Szczecin University of Technology – Research and Education Center of Modern Multimedia Technologies" (Poland).

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X GPU used for this research.

References

1. Astrand, B., Baerveldt, A.: A vision-based row-following system for agricultural field machinery. Mechatronics **15**(2), 251–269 (2005)
2. Blackman, S., Popoli, R.: Design and Analysis of Modern Tracking Systems. Artech House, Norwood (1999)
3. Deans, S.R.: The Radon Transform and Some of Its Applications. Wiley, New York (1983)
4. Golightly, I., Jones, D.: Visual control of an unmanned aerial vehicle for power line inspection. In: 12th International Conference on Advanced Robotics, ICAR 2005, pp. 288–295, July 2005
5. Haykin, S., Moher, M.: Communication Systems. Wiley, Chichester (2009)
6. King, D.E.: Dlib-ml: a machine learning toolkit. J. Mach. Learn. Res. **10**, 1755–1758 (2009)
7. LeCun, Y., Kavukcuoglu, K., Farabet, C.: Convolutional networks and applications in vision. In: Proceedings of 2010 IEEE International Symposium on Circuits and Systems, pp. 253–256, May 2010
8. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015). http://dx.doi.org/10.1038/nature14539

9. Matczak, G., Mazurek, P.: History dependent viterbi algorithm for navigation purposes of line following robot. Image Process. Commun. **20**(4), 5–11 (2016)
10. Mazurek, P.: Hierarchical track-before-detect algorithm for tracking of amplitude modulated signals. In: Choraś, R.S. (ed.) Image Processing & Communications Challenges 3. AISC, vol. 102, pp. 511–518. Springer, Heidelberg (2011). doi:10.1007/978-3-642-23154-4_56
11. Mazurek, P.: Line estimation using the viterbi algorithm and track-before-detect approach for line following mobile robots. In: 2014 19th International Conference on Methods and Models in Automation and Robotics (MMAR), pp. 788–793, September 2014
12. Mazurek, P., Putynkowski, G.: Frequency management for electromagnetic continuous wave conductivity meters. Sensors **16**(4), 490 (2016)
13. Mazurek, P.: Directional filter and the Viterbi algorithm for line following robots. In: Chmielewski, L.J., Kozera, R., Shin, B.-S., Wojciechowski, K. (eds.) ICCVG 2014. LNCS, vol. 8671, pp. 428–435. Springer, Cham (2014). doi:10.1007/978-3-319-11331-9_51
14. Ollis, M.: Perception Algorithms for a Harvesting Robot. CMU-RI-TR-97-43, Carnegie Mellon University (1997)
15. Scott, T.A., Nilanjan, R.: Biomedical Image Analysis: Tracking. Morgan & Claypool, San Rafael (2005)
16. Stone, L., Barlow, C., Corwin, T.: Bayesian Multiple Target Tracking. Artech House, Norwood (1999)
17. Viterbi, A.: Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Trans. Inf. Theory **13**(2), 260–269 (1967)
18. Wang, R.: Edge detection using convolutional neural network. In: Cheng, L., Liu, Q., Ronzhin, A. (eds.) ISNN 2016. LNCS, vol. 9719, pp. 12–20. Springer, Cham (2016). doi:10.1007/978-3-319-40663-3_2

Properties of Genetic Algorithms for Automated Algebras Generation

Hashim Habiballa[✉], Matej Hires, and Radek Jendryscik

Department of Informatics and Computers, University of Ostrava,
30. dubna 22, Ostrava 1, Czech Republic
hashim.habiballa@osu.cz

Abstract. The article describes research concerning properties of genetic algorithms for automated generation of specific algebras. We provide brief explication of the task and the main focus is directed towards properties of the designed approach and their statistical significance. Main results include the impact of the mutation operator and the notion of "colourfulness", which defines the practical usage of automatically designed population members.

1 Introduction

In many situations in mathematics and theoretical computer science there is a need for special structures - algebras with elements, operations and axioms to be fulfilled. Even for finite structures it may be very hard task for a mathematician to design such operation definitions in order to satisfy complex properties.

The natural candidate for this purpose we can see in evolutionary techniques that are well-proved methods for tasks requiring enormous state-space searching [6,7]. There is also a "fitness criterion" concerning fulfilment of several compulsory and optional axioms by any candidate structure. Therefore we tried to design, implement and test this approach which proved to be effective in contrast to standard state-space searching.

We have developed a software tool called EQCreator, which works on the principles of genetic algorithms and is able to produce EQ-algebras (truth value structures based on equality) in reasonable time. We also show results of preliminary experiments with various additional algebra operators and special axioms leading to promising EQ-algebras with interesting properties suitable for fuzzy logics.

The article outlines our experiments concerning properties of the evolutionary approach with the focus to the mutation rate and colourfulness of generated algebras.

2 Finite Algebras Automated Production

The problem we are solving can be formulated as follows. We would like to design and create **finite algebras with specific properties:**

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_42

- n – number of algebra elements (finite)
- Algebra operations declaration
- Compulsory properties of operations (axioms)
- Optional properties of operations (axioms)
- Generate such algebra fulfilling requirements given by axioms

It is clear that even for very small algebras (low number of structure elements) the extent of state space is enormous and it forms unrealistic number of candidates. This superexponential complexity prevents us to use standard state-space searching algorithms and one of the natural possibilities focuses us into evolutionary techniques.

3 Genetic Algorithms

Genetic algorithms (GA) provide proved methods for automated design of optimal structures based on evolution inspired procedures [4,5]. Despite its simple principle it is a challenge to find suitable settings of parameters and types of crossover and mutation. Among sporadic papers concerning particular application of GA for automated production of algebras there is an interesting paper [1]. Although it describes results and overall settings used for Genetic programming, it lacks details concerning used crossover and mutation algorithms. There is also very high level of mutation (in some cases above 50%!) that shows problems with convergence of the process. We followed another method which uses pure GA and we will also try to compare our results with [1]. But the reader should consider our approach and objective are different. Nevertheless some of our results conform to the cited ones.

Main characteristics of Genetic algorithms:

- Population member (candidate solution), its fitness function (evaluates suitability)
- Population – set of members, starting population (random)
- New generation created from previous by selection, crossover and mutation
- Generate new populations until stop condition is fulfilled (fix number of iterations – populations, predefined member fitness being optimal etc.)

Population and Population Member (GA)

- Candidate solution p (Population Member/PM) represented by its properties (usually stored in “chromosomes” – bit array, integer array etc.)
- Fitness function of candidate solution f, $f(x) \in \langle 0, 1\rangle$, x is PM – the keystone of time complexity of the task (possible parallelism)
- Population – fix or variable number of PM: Population member (candidate solution), its fitness function (evaluates suitability), Population – sets of PMs, best PM, worst PM, median PM, Generation – sequence of populations called generations $G_0, \ldots, G_r$, where $G_i = \{p_{i,j} | i, j \in N\}$, i is generation index, j is PM index in population,
- Starting Generation G_0 is randomly (partially randomly) generated.

Genetic operators (GA)

- Selection – simply into next generation or further processing: Elitist – usually best m PM from G_i is directly copied into G_{i+1}, Selection for crossover (SC) – some PMs from G_i are selected for generation of new children for G_{i+1}, SC should inhere probability of selection $prob_{SC}(p)$ for PM p non-decreasing with respect to fitness function: $f(p_1) \geq f(p_2) \Rightarrow prob_{SC}(p_1) \geq prob_{SC}(p_2)$.
- Crossover – combination of several PMs to generate new PMs for next generation: Simple – two old PMs p_{old1}, p_{old2} generate two children, where first portion of chromosome is from p_{old1} and second from p_{old2} and contrary, Exponential – if we can distinguish several portions of chromosome we can generate more children than parents (every possible combination).
- Mutation – randomly selected PMs from new generation are "altered": Mutation rate – probability of selection PM for mutation, Point – single element of chromosome is altered, Interval – interval of chromosome elements are altered, Overall – whole chromosome is altered.

4 EQ-algebras

Our task was to generate specific algebras – **EQ-algebras**. EQ-algebras serve as truth value structure for EQ-logics [2], which form current studied fuzzy logics in the field of fuzzy logic research [3]. Instead of implication, their key operation is Fuzzy Equality. EQ-algebra has three basic operations in total: Infimum $\wedge$, Multiplication $\otimes$, Fuzzy Equality $\sim$. There are also derived additional supporting (directly following) operations – Implication $\rightarrow$, Negation $\neg$, and relational operator LessThanOrEqual $\leq$.

EQ-algebra $\mathcal{E}$ is algebra of type (2, 2, 2, 0), i.e.

$$\mathcal{E} = \langle E, \wedge, \otimes, \sim, \mathbf{1} \rangle \tag{1}$$

(E1) $\langle E, \wedge, \mathbf{1} \rangle$ is a commutative idempotent monoid (i.e. $\wedge$-semilattice with top element $\mathbf{1}$). We put $a \leq b$ iff $a \wedge b = a$, as usual.

(E2) $\langle E, \otimes, \mathbf{1} \rangle$ is a monoid and $\otimes$ is isotone w.r.t. $\leq$.

(E3) $a \sim a = \mathbf{1}$ (reflexivity axiom)

(E4) $((a \wedge b) \sim c) \otimes (d \sim a) \leq c \sim (d \wedge b)$ (substitution axiom)

(E5) $(a \sim b) \otimes (c \sim d) \leq (a \sim c) \sim (b \sim d)$ (congruence axiom)

(E6) $(a \wedge b \wedge c) \sim a \leq (a \wedge b) \sim a$ (monotonicity axiom)

(E7) $a \otimes b \leq a \sim b$ (boundedness axiom)

5 Specific Genetic Algorithms for EQ-algebras Design

In order to generate candidate EQ-algebras for further research we utilized GA under specific settings. Implementation is done by object oriented model of EQ-algebras as GA Population Members. GA Population (Generation) is implemented as *list* of PMs. Fitness function is based on relative fulfilment of mandatory and optional axioms. EQ-algebras fulfilling additional criteria are called Winners and they are stored during GA process. We have to note that very important issue is detection of previously generated (identical) candidates (removal). Random (starting) population is partially built to fulfil simple properties (e.g. infimum is commutative). Fitness evaluation has two phases:

- Mandatory properties evaluation (e.g. boundedness axiom – $a \otimes b \leq a \sim b$)
- Optional properties evaluation (e.g. goodness – $a \sim \mathbf{1} = a$)

In every generation we perform sorting of PMs in population through fitness. Termination condition is currently based on:

- Fixed number of steps performed
- Fixed number of EQ-algebras with required properties
- Manual (user) termination

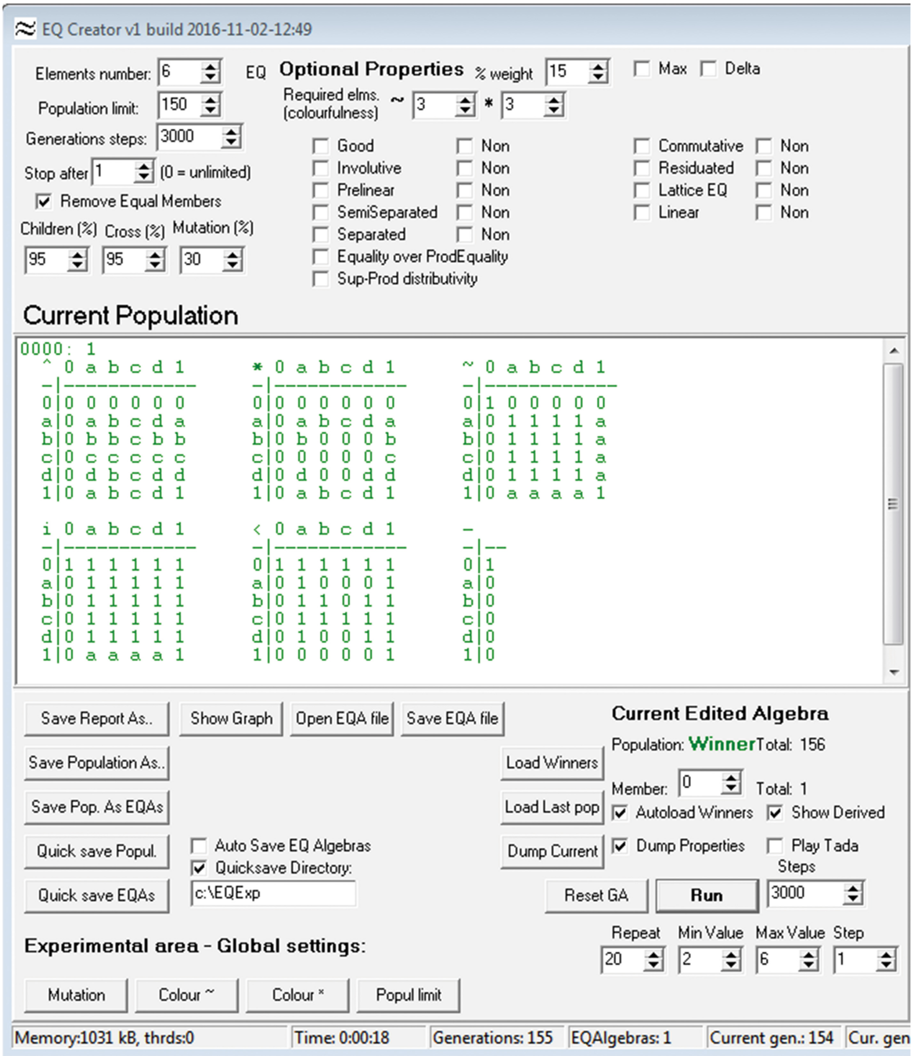
Algorithms are implemented in the form of PC application EQCreator – GUI based application for MS Windows 32-bit platform. Its main purpose is following:

- Selection of various properties for candidate EQ-algebras
- Evolution of algebras to attain EQ-algebras even with specific properties
- Automated check of properties and generation
- Saving of resulting optimal solutions in suitable form

It enables to set mainly – Algebra elements number – support size (2–28), Population limit – max. number of algebras in population, Generation steps – max. number of GA steps until one run stops (except stopped manually) (0 - unlimited) and Stopping after certain number of EQ-algebras found. The variability of GA is also assured by the possibility of setting basic GA properties:

- Children ratio (0–100%) – crossover resulting new members relative count (how large portion of new population to be new children, others are old members copied from previous generation)
- Cross ratio (0–100%) – portion of BEST members to have possibility to crossover (it is not crossover probability!)
- Mutation ratio (0–100%) – probability for new population member to be mutated
- Crossover probability is set arbitrary (fixed) – in descending ordered (by fitness) population of the size N we set probability of member i $p_i = \frac{N-i}{\frac{N*(N+1)}{2}}$ for $i = 0, \ldots, N-1$, where $f(i) \geq f(i+1)$ (fitness for members)
  e.g. for 5 members: $p_0 = \frac{5}{15}, p_1 = \frac{4}{15}, ..., p_4 = \frac{1}{15}$
- weight of optional properties – relative weight of special EQ-algebras requirements (e.g. linear EQA, involutive EQA) – should be significantly less than for compulsory axioms (experimental best – 15%)
- notion of colourfulness – required number of distinct elements in variable positions for operator function values (some combinations are determined e.g. $a \wedge 0 = 0$ in every EQA)

- **colourfulness** assures non-trivial EQ-algebras to be generated e.g. for fuzzy equality when 3 of 5 required - at least 3 different elements occur as functional values in non-determined cases
- colourfulness experimentally needed for Multiplication ($\otimes$) and Fuzzy Equality ($\sim$) – higher means computationally harder! (Fig. 1)



EQ Creator v1 build 2016-11-02-12:49

Elements number: 6
Population limit: 150
Generations steps: 3000
Stop after 1 (0 = unlimited)
☑ Remove Equal Members
Children (%) 95 Cross (%) 95 Mutation (%) 30

EQ **Optional Properties** % weight 15
Required elms. (colourfulness) ~ 3 * 3

□ Max □ Delta

□ Good □ Non
□ Involutive □ Non
□ Prelinear □ Non
□ SemiSeparated □ Non
□ Separated □ Non
□ Equality over ProdEquality
□ Sup-Prod distributivity

□ Commutative □ Non
□ Residuated □ Non
□ Lattice EQ □ Non
□ Linear □ Non

Current Population

```
0000: 1
  ^ 0 a b c d 1      * 0 a b c d 1      ~ 0 a b c d 1
  -|------------     -|------------     -|------------
  0|0 0 0 0 0 0      0|0 0 0 0 0 0      0|1 0 0 0 0 0
  a|0 a b c d a      a|0 a b c d a      a|0 1 1 1 1 a
  b|0 b b c b b      b|0 b 0 0 0 b      b|0 1 1 1 1 a
  c|0 c c c c c      c|0 0 0 0 0 c      c|0 1 1 1 1 a
  d|0 d b c d d      d|0 d 0 0 d d      d|0 1 1 1 1 a
  1|0 a b c d 1      1|0 a b c d 1      1|0 a a a a 1

  i 0 a b c d 1      < 0 a b c d 1      -
  -|------------     -|------------     -|--
  0|1 1 1 1 1 1      0|1 1 1 1 1 1      0|1
  a|0 1 1 1 1 1      a|0 1 0 0 0 1      a|0
  b|0 1 1 1 1 1      b|0 1 1 0 1 1      b|0
  c|0 1 1 1 1 1      c|0 1 1 1 1 1      c|0
  d|0 1 1 1 1 1      d|0 1 0 0 1 1      d|0
  1|0 a a a a 1      1|0 0 0 0 0 1      1|0
```

□ Auto Save EQ Algebras
☑ Quicksave Directory: c:\EQExp

Current Edited Algebra
Population: **Winner** Total: 156
Member: 0 Total: 1
☑ Autoload Winners ☑ Show Derived
☑ Dump Properties □ Play Tada
Steps 3000

Experimental area - Global settings:
Repeat 20 Min Value 2 Max Value 6 Step 1

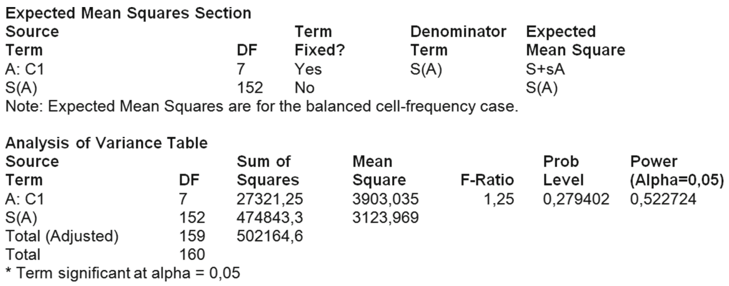
Memory:1031 kB, thrds:0 | Time: 0:00:18 | Generations: 155 | EQAlgebras: 1 | Current gen.: 154 | Cur. gen

Fig. 1. EQCreator and example EQ-algebra

6 Experimental Evaluation and Optimization

Even the above mentioned approach made the process of finite algebras generation possible, it is sensitive to proper settings of the genetic algorithm. The main problem discovered already in [1] is connected with the level of mutation ratio - probability (the probability of a new population member to be mutated). The high probability has been observed in the article and it is also our experience during preliminary experiments. As it is a keystone of the genetic algorithms together with suitable crossover implementations, we have been thoroughly experimenting with this parameter.

We have created special version of EQCreator software for experiments, in order to produce statistically appropriate samples to test the measurement results with statistical software NCSS.



```
Expected Mean Squares Section
Source                                  Term        Denominator  Expected
Term                            DF      Fixed?      Term         Mean Square
A: C1                           7       Yes         S(A)         S+sA
S(A)                            152     No                       S(A)
Note: Expected Mean Squares are for the balanced cell-frequency case.

Analysis of Variance Table
Source                         Sum of       Mean                       Prob        Power
Term                   DF      Squares      Square          F-Ratio    Level       (Alpha=0,05)
A: C1                  7       27321,25     3903,035           1,25    0,279402    0,522724
S(A)                   152     474843,3     3123,969
Total (Adjusted)       159     502164,6
Total                  160
* Term significant at alpha = 0,05
```

Fig. 2. Analysis of variance test on mean time - experiment 1

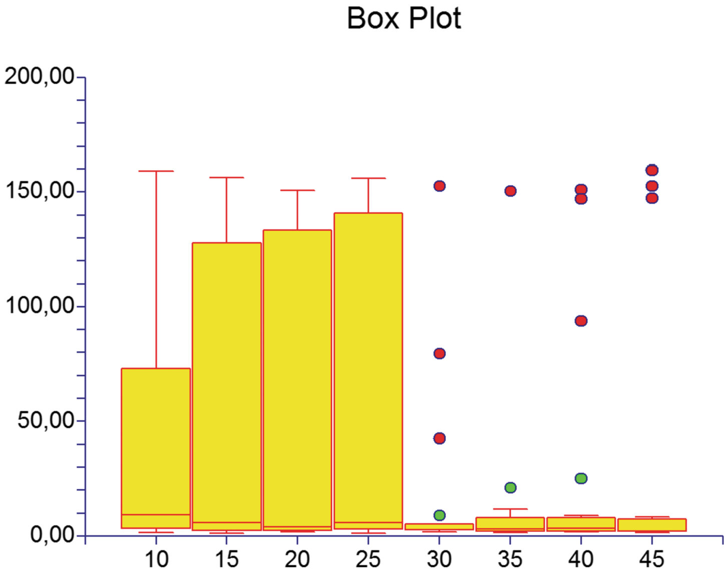


Fig. 3. Analysis of variance box plot - experiment 1

6.1 Experiment No. 1

- algebra elements $n = 6$
- mutation rate from 10% to 45% step 5%
- no special algebra constraints
- colourfulness for multiplication ($\otimes$) at level 3 and for fuzzy equality ($\sim$) at level 3
- Time (seconds) vs. mutation rate, 20 samples each 8 rates

Analysis of variance results

Analysis of variance for time vs. mutation rate shows no statistically significant differences (Fig. 2), but the plot of the means show interesting decrease in measured time with best results in 30–35% and then again worsening results with increase in mutation rate Figs. 3 and 4.

Fisher's LSD Multiple-Comparison Test shows statistically significant (0,05) difference between level 35% and 25% (Fig. 5).

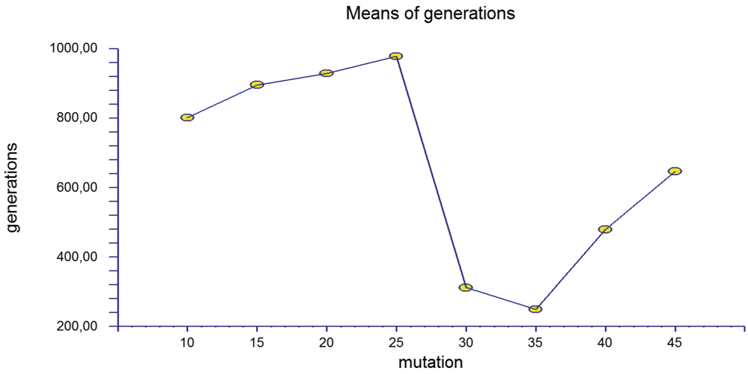
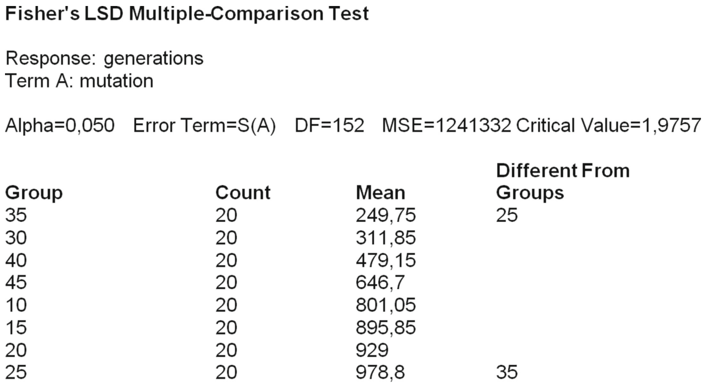


Fig. 4. Means of generations - experiment 1



Fisher's LSD Multiple-Comparison Test

Response: generations
Term A: mutation

Alpha=0,050 Error Term=S(A) DF=152 MSE=1241332 Critical Value=1,9757

Group	Count	Mean	Different From Groups
35	20	249,75	25
30	20	311,85	
40	20	479,15	
45	20	646,7	
10	20	801,05	
15	20	895,85	
20	20	929	
25	20	978,8	35

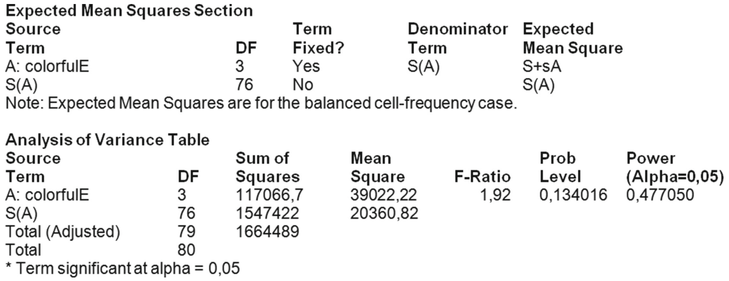
Fig. 5. Fisher's LSD multiple comparison test of generations - experiment 1

6.2 Experiment No. 2

- algebra elements $n = 6$
- colourfulness for fuzzy equality ($\sim$) from 2 to 5 step 1 (note that colourfulness for 6 elements has no meaning due algebra properties)
- no special algebra constraints
- mutation rate set at 30%
- Time (seconds) vs. mutation rate, 20 samples each 4 rates

Analysis of variance results

Analysis of variance for time vs. colourfulness shows no statistically significant differences. There is natural increase in computed time due to increasing



Expected Mean Squares Section

Source Term	DF	Term Fixed?	Denominator Term	Expected Mean Square
A: colorfulE	3	Yes	S(A)	S+sA
S(A)	76	No		S(A)

Note: Expected Mean Squares are for the balanced cell-frequency case.

Analysis of Variance Table

Source Term	DF	Sum of Squares	Mean Square	F-Ratio	Prob Level	Power (Alpha=0,05)
A: colorfulE	3	117066,7	39022,22	1,92	0,134016	0,477050
S(A)	76	1547422	20360,82			
Total (Adjusted)	79	1664489				
Total	80					

* Term significant at alpha = 0,05

Fig. 6. Analysis of variance test on mean time - experiment 2

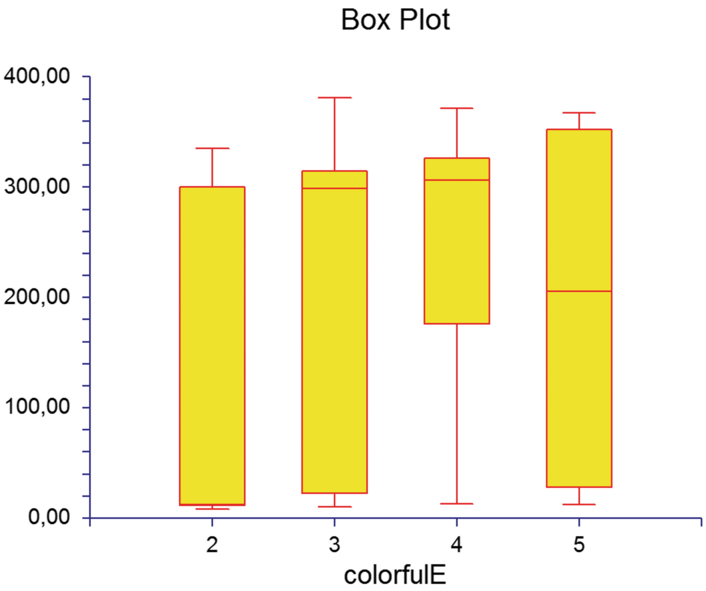


Fig. 7. Analysis of variance box plot - experiment 2

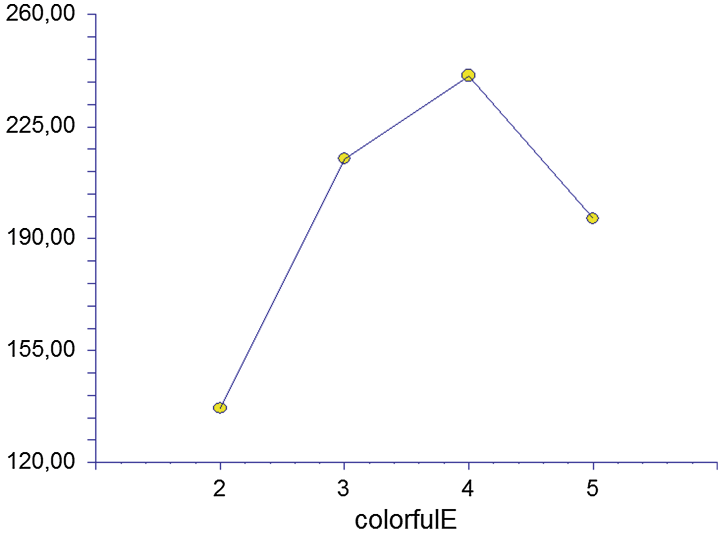


Fig. 8. Analysis of variance means plot - experiment 2

demands on colourfulness of fuzzy equality. Last value shows a slight decrease, but not statistically significant (Figs. 6, 7 and 8).

7 Conclusion

The experimental evaluation provided us with important results showing the importance of mutation rate and colourfulness. Although the mutation ratio has to be set unusually high in contrast to standard applications of genetic algorithms, it is **significantly lower** than in previous works of other authors [1]. There is statistically significant difference around 30% compared to other ratios computed. Furthermore we observed the differences between colourfulness levels are not statistically significant (despite its variable computational complexity). It enables us to produce more interesting and promising algebras without significantly affecting effectiveness of the evolutionary generation process.

We would like also work in future especially on constrained EQ-algebras since the evolution of these algebras is even more computationally hard task. There is also very interesting the usage of parallel computations, i.e. using of either multicore or multiprocessor computing or high performance vector EUs of graphic cards. The latter option nevertheless will require complex redefinitions and optimizations of algorithms.

References

1. Spector, L., et al.: Genetic programming for finite algebras. In: Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation (GECCO 2008), NY, USA, pp. 1291–1298 (2008)
2. Novák, V., De Baets, B.: EQ-algebras. Fuzzy Sets Syst. **160**, 2956–2978 (2009)
3. Dyba, M., Novák, V.: EQ-logics: non-commutative fuzzy logics based on fuzzy equality. Fuzzy Sets Syst. **172**, 13–32 (2011)
4. Hingston, P., Barone, L., Michalewicz, Z.: Design by Evolution: Advances in Evolutionary Design. Springer, Heidelberg (2008). ISBN 978-3540741091
5. Volna, E., Kotyrba, M.: A comparative study to evolutionary algorithms. In: Proceedings 28th European Conference on Modelling and Simulation, ECMS 2014, Brescia, Italy, pp. 340–345 (2014)
6. Sekanina, L.: Evolvable hardware. In: Handbook of Natural Computing. Springer, Heidelberg, pp. 1657–1705 (2012). ISBN 978-3-540-92909-3
7. Kotyrba, M., Volna, E., Bujok, P.: Unconventional modelling of complex system via cellular automata and differential evolution. Swarm Evol. Comput. **25**, 52–62 (2015). Elsevier, Amsterdam. ISSN 2210–6502

An Application of a Discrete Firefly Algorithm in the Context of Smart Mobility

Ezzeddine Fatnassi(✉), Noor Guesmi, and Tesnime Touil

Institut Supérieur de Gestion de Tunis, Université de Tunis, 2000 Le Bardo, Tunisia
ezzeddine.fatnassi@gmail.com

Abstract. Firefly algorithm (FA) is a swarm intelligence based optimization method. It is based on the social behavior of fireflies where the brighter firefly attracts the less brighter one. In this paper, we present a discretization of FA in order to solve a problem related to the context of smart mobility. First, the context of smart cities and smart mobility is presented and a related optimization problem is proposed. Then, a discretization of FA to solve the problem is proposed. The proposed algorithm is based on transforming the FA functions into a discrete ones capable of manipulating a permutation of integers. Computational experiments on a set of instances from the literature demonstrate the efficiency of the proposed methodology.

Keywords: Smart cities · Smart mobility · Firefly algorithm · Electric vehicle

1 Introduction

Nowadays, the world is evolving at an incredible pace. In fact and in 2008, the United Nation stated that around 50% of all people are living in urban areas. Forecasts also show that in 2050 70% of the total population will live in urban areas. Consequently, cities centers are becoming the hub and the center of the physical (telecommunication and technologies) and human (qualified staff) capitals as well as commerce. This is due to the transformation observed by the world economy as it is becoming more integrated and service based. Cities are also considered as the majors centers for resource consumption. It should be noted that urban areas are responsible for 75% of the world's energy consumption and 80% of the greenhouse emission.

These features contribute on creating new urban environment characterized by the growing demand for the increase of the urban's quality of life as well as a sustainable and efficient development and an efficient use of the resources. Consequently, new ways of managing cities need to be considered in order to optimize resources [15].

Based on this context, research in smart cities and Intelligent Transportation Systems is needed in order to solve problems related to urban areas in a smarter ways.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_43

Smart city can be defined as the city that uses Information and Communications Technology (ICT) in order to offer to its citizen an interactive and efficient infrastructure and utilities. Consequently, ICT is considered as a key factor for the transition from basic infrastructure to a more connected and sustainable infrastructures enabling a smart city.

There are six most-common pillars of smart cities: (i) smart economy, (ii) smart people, (iii) smart governance, (iv) smart mobility, (v) smart environment, and (vi) smart living [7]. There are different ways of enabling a city to be smart. Among them one could consider the implementation of new services based on city's priorities. The implementation of such services has to take into account not only local regulation and priorities but also the citizens preferences as they are the main target of such services. In this paper, we focus on the reduction of the total travel time of citizens (Smart Mobility) and the reduction of greenhouse gas emission (Smart Environment). These two elements of the smart cities concept fuel the need for the implementation of public and smart urban transportation tools that reduce the impact of transit options on the environment.

Therefore, it comes the need for a novel and sophisticated transportation system that is efficient in term of ecological and societal aspect that could improve and overcome many issues related to urban mobility of goods and persons. Generally in smart cities and smart mobility, there are two ways to deal with sustainability issue:

- Enhance the quality and the performance of already existing urban movers' tools.
- Invest in a completely new and enhanced transportation system that is more ecological and energy efficient than the traditional and conventional transportation tool.

In this paper, we focus on the second option as we aim to introduce a relatively new transportation system within the concept of smart cities namely Personal Rapid Transit (PRT). PRT is an innovative transportation tool as it operates a set of electric driverless vehicles over a network of dedicated guideways.

In order to minimize its total energy consumption a discrete Firefly Algorithm is proposed to tackle a static deterministic routing problem related to PRT.

The proposed PRT problem is an optimization one which is proven to be NP-hard. Hence, it is difficult to be solved using a traditional exact method within small computational time which represents the requirement for such a solution approach in a highly dynamic context. Therefore, it comes the need for an effective algorithms to tackle the proposed problem. In the recent years, several new optimization algorithms inspired by swarm intelligence have been proposed e.g. ant colony optimization (ACO) [14], artificial bee colony (ABC) [8] and Firefly Algorithm (FA) [16].

FA is a bio-inspired optimization algorithm that mimic the social behavior of fireflies. Fireflies use light in order to attract other mates. Attraction among fireflies determines their movements. The attractiveness is related to the brightness of fireflies. Since its introduction, FA has been applied to several optimization problems such as nonlinear and non-convex optimization problems [16,17].

FA is implemented initially for solving continuous optimization problems. In case we want to apply the FA to a discrete problem such as the PRT, modifications need to be made on the original FA in order to make it able to solve the proposed PRT problem.

In this paper, we propose a FA variant that simulates the behavior of fireflies behavior by implementing discrete versions of the different components of FA. Consequently, the main contributions of this paper can be summarized as follows. First, a discrete version of FA is proposed in order to be applied in the context of smart mobility and green transportation. Second, the computational time of the proposed algorithm is analyzed. Third, the proposed approach is shown to effectively reduce the total travel time of PRT system.

In this paper, a methodology for modeling a static problem related to PRT will be presented. Then, a discrete FA is proposed to tackle this problem. Next, the computational results are described. Finally, we provide some conclusions and perspectives.

2 PRT Problem Definition and Modeling

Integrating PRT into a smart city would be very beneficial in the context of smart mobility and smart environment. In fact, the system has the possibility to offer several advantages over classical transportation tools in order to reduce total travel time, congestion and carbon emissions. In the literature, several studies [5] already show that the total trip time and carbon emissions using PRT system are less than several public and classic transportation tools.

Based on the specificities of PRT as a potential improvement in the context of smart cities, we could note that it is important to focus on the economic and environmental urban context while implementing such an innovative system. In fact, the PRT offers an on-demand transportation service which results necessarily in a set of empty vehicles displacements. More specifically as the demand ending at a specific station does not necessarily equal to the demand starting from that same station, the PRT system could generate a set of displacements where vehicles move empty in order to take specific passengers from a station. This set of empty moves engenders an economic loss as well as contributing on harming the environment as the energy used for the empty displacement was simply wasted and not used efficiently. So it could be interesting from an operational point of view to enhance its efficiency by reducing its empty moves while making it more sustainable and economically reliable.

Consequently, we propose in the next to solve a static routing problem related to PRT [12]. Let us suppose to have a fully connected PRT network with a set of M stations and one depot named D. We focus in this paper exclusively on the operational use of PRT.

Consequently, we suppose to have a list of PRT' transportation demand T that need to be satisfied. Each demand $i \in T$ is characterized by a quadruplet: (i) DT_i: depart time, (ii) AS_i: arrival station, (iii) DS_i: depart station, (iv) AT_i: arrival time which is the is the depart time DT_i plus the duration of the trip with shortest path, between the depart and the arrival stations.

We suppose also to have a set of unlimited number of vehicles initially located in D. Each vehicle has a limited battery capacity denoted B that make it run for a limited determined time. PRT batteries could only be charged at the depot. We suppose also to have a matrix named Sp representing the shortest path between each pair of stations.

The proposed problem is modeled based on a graph $G = V, E$. The set of nodes is represented by V. Each node in V represents one PRT' transportation demand. Two dummy nodes s and t are added to V which represent the depot. $V^* = V/\{s,t\}$ Let us define also E as the set of arcs. $E^* = E/\{(i,j); i = s$ or $j = t\}$ where s and t are two dummy nodes representing the depot. For each pair of nodes i and j, we define an arc based on the following rules:

- for each i and $j \in V^*$ and $AT_i + Cost_{(AS_i, DS_j)} \leq DT_j$ an arc (i,j) is added with a cost c_{ij}, equals to the travel time needed to move from AS_i to DS_j in addition to the travel time needed to move from DS_j to AS_j
- We add a set of arcs connecting the depot node s to each node in V^*. The cost of this arc is equal to the travel time needed to reach the arrival station of node i from the depot while passing by the departure station of node i.
- for each node $i \in V^*$, we add an arc (i,t). The cost of this arc is the travel time needed to reach the depot starting from the arrival station of trip i.

The objective of our problem is to find the set of roads starting and ending at the depot while covering all the trip nodes exactly once and respecting the battery capacity of the PRT' pods. Based on this problem modeling, we could note that our problem is assimilated to the asymmetric distance constrained vehicle routing problem (ADCVRP). The ADCVRP is the problem of finding a set of vehicles'roads starting and ending at the depot with respect to a maximum allowable distance constraint. The ADCVRP is proven to be NP-Hard [1]. This theocratical problem was studied in the literature in the work of Laporte et al. [9] and in the work of Almoustafa et al. [1]. The high complexity of this problem (NP-Hard) makes it suitable to be solved using a bio-inspired algorithm [6]. In the next, we focus on adapting and applying a discrete fire flies algorithm for solving the proposed problem.

3 Firefly Algorithm

3.1 Basic Firefly Algorithm

Firefly algorithm was developed first by Xin-She Yang [16]. FA is like the particle swarm optimization (PSO) algorithm. It is a stochastic based search algorithm which manipulates a population of firefly. Solution of the proposed problem is represented by a single firefly in the population. Fireflies generally move to other positions in the search space in order to find potential new solutions. Attractiveness between different fireflies is determined by the intensity of the emitted light. The emitted light is proportional to the fitness value of the firefly.

Algorithm 1. Basic Firefly Algorithm

```
Initialize the population of Fireflies of size $n$
while (termination criterion is not met) do
    for ($i = 0$ to $n$) do
        for ($j = 0$ to $n$) do
            if $F(x_i) > F(x_j)$ then
                Move $x_i$ toward $x_j$
                $d_{ij} \leftarrow$ Distance($x_i$,$x_j$)
                $\beta \leftarrow attractivness(I_0, \gamma, d_{ij})$
                $x_i \leftarrow (1 - \beta)x_i + \beta x_j + \alpha(Random() - \frac{1}{2})$
                Compute the fitness value of the new $x_i$.
            end if
        end for
    end for
    Best firefly moves randomly
end while
```

The attractiveness of a firefly is attenuated over the distance. The basic form of FA is presented in Algorithm 1.

In Algorithm 1, n is the size of the population. I_0 is the intensity of light at the source. α represents the size of the random step and γ represents the absorption coefficient.

3.2 FA Discretization for PRT Problem

Based on Algorithm 1, it is clear that several functions related to FA need to be re-implemented in order to be able to solve the PRT problem such as the steps of movements of the fireflies as well as the attraction function.

To do so, solutions (the set of fireflies) in our algorithm are represented by a permutation of trips and the algorithm manipulates them accordingly. To evaluate the different solutions in our algorithm, a cost function from the vehicle routing problem (VRP) literature is adapted. In fact, we use the Split function based on the work of Prins [13]. This cost function and based on a permutation of customers in VRP could find a set of minimum cost routes. An auxiliary graph is constructed where each node represents a VRP customer and each edge represents a feasible route based on the permutation of customers. Then, the Bellman algorithm [2] is used in order to find the least cost set of routes. The same principle is applied in our algorithm where a solution is represented by a set of trips instead of a set of customers. More details could be found in [13].

3.3 Initial Fireflies

In the FA version proposed by Xin-She Yang [16], the initial fireflies are generated randomly in order to cover the whole search space in an uniform distribution. Several approach in the literature exists in order to generate initial solutions for optimization problems such as greedy approaches. However and in this paper, we propose to use random solution'generation as using this method would guarantee to have a scattered initial population over the search space and therefore a more diverse fireflies. We added to the initial population one enhanced firefly based on the constructive heuristic proposed by Mrad et al. [11].

3.4 Distance Function

In the literature, there is different ways to compute the distance between two distinctive solutions:

- The Hamming distance which corresponds to the total number of trips minus the number of trips that exist in the same position on the two solutions
- The Swap distance which corresponds to the number of swaps needed to move from the first to the second solution.

However, it is clear that the Hamming distance is proportional to the difference in the objective function between the two compared solutions. Consequently and based on this observation, we compute in this paper the difference of the objective function using the Split function [13] in order to measure the distance between two different solution.

3.5 Attraction Function

The implementation of the attraction between two different fireflies represents the one of the most important feature to be implemented in order to apply the FA in the context of smart mobility. The FA was initially implemented for continuous optimization. Therefore, the attraction function needs to be implemented in a way to respect its original implementation for continuous optimization.

The original FA algorithm and in order to compute the next position of a firefly use the following function:

$$x_i \leftarrow (1-\beta)x_i + \beta x_j + \alpha(Random() - \frac{1}{2}) \tag{1}$$

Equation 1 could be divided into two subfunctions

$$x_i \leftarrow (1-\beta)x_i + \beta x_j \tag{2}$$

and

$$x_i \leftarrow x_i + \alpha(Random() - \frac{1}{2}) \tag{3}$$

Therefore and in order to implement a discrete version of FA, we need to implement a search versions of the Eq. 2 (which we call the β step) and 3 (which we call the α step).

3.6 The β Step

The β step should be implemented in such a way to bring one firefly closer to another firefly. Consequently and based on this principle, the distance between two solutions should be reduced after applying the β step. Let us recall that we are using the difference between the fitness of two solutions as a measure of their relative distance. Therefore and in order to implement the β step, the set'size of common trips between the two solution needs to increase. Consequently and as a first step, we need to extract the set of the common trips between the two solutions and insert it into the output solution. Figure 1 shows an example of this first step.

Next, we need to fill the gaps in the output solution using the rest of missing trips from the two input solutions i and j. Therefore, we need to iterate on the gaps in the resulting solution and choose whatever to insert a trip from the solution i or j. We need also to ensure that there is no redundant trips in the resulting solution as we are working on a permutation as a representation of the solutions in the FA. This could be done using a probability $\beta = \frac{1}{1+\gamma d_{ij}}$ where d_{ij} is the distance between solutions i and j and γ is a parameter for the FA which defines the amount of randomness in the selection between the two solutions. Finally, we need to fill the gaps in the resulting solution in a random order to guarantee a minimum of diversity in the output solution. An illustrative example is shown in Fig. 1. In the first step, we copied the trip from solution j in order to fill the gap at position 6. Next, we copied a trip from solution i in order to fill the gap at position 2. In the third step, we did not copy any trips at position 5 as the two trips from solutions i and j are already used in the output solution.

Finally in the β step, we need to fill the last gaps in the output solution with the missing trips in a random order in order to obtain a valid solution. Therefore and based on the example shown in Fig. 1, the only missing trip is trip 8. Consequently, we insert it in the remaining gap in the output solution.

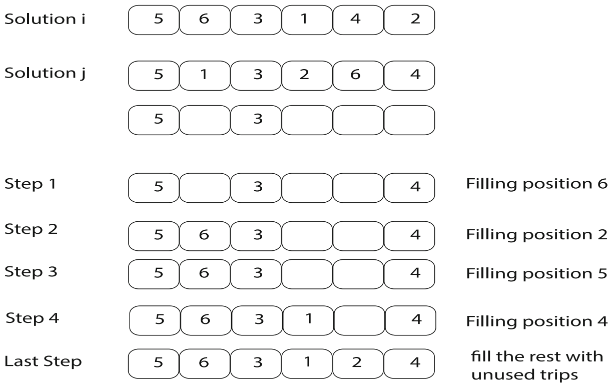


Fig. 1. Illustrative example of the β Step.

3.7 The α Step

The α step is implemented in this paper as a shift to move to one of the neighboring solution of its input solution. Therefore, the α step could be implemented in a variety of ways. However, we should keep in mind that the result solution of the α step needs also to mimic the attraction process of the fireflies. In this paper and to do so, we applied several random swaps in the input solution in order to obtain the new generated firefly. Using swaps as neighborhood operator guarantees to generate a solution with the minimum distance between the input and output solution.

4 Experimental Results

We now present the computational study performed to investigate the results of our FA for solving the proposed optimization problem. An analysis of the obtained results is also performed to prove the efficiency of our algorithm.

4.1 Test Instances

The FA is implemented using a personal computer with an Intel i5 3.2 GHz CPU and 8 GB of RAM, and running the Microsoft Windows 7 operating system. Since we are treating a PRT routing problem, we adapted instances from the literature to our context in order to test the FA [3]. The instances used in this paper are based on the number of trips to serve which varies between 10 and 100 trips in a step of 5. For each size of trips, we used 40 instances. For more details about the PRT instances and the instance generator the reader is refereed to [12].

4.2 Computational Results

The computational results of the FA are exposed in Table 1. The obtained results are compared against a valid lower bound values (LB) [4] taken from the literature for the proposed problem using the GAP metric which is computed as follows.

$$\text{GAP} = (\frac{(SOL - LB)}{LB}) \times 100$$

Table 1 shows that the proposed FA obtains good quality results for the proposed optimization problem. More specifically, the FA founds an average gap of 2.621%. Also, we could note that the average GAP varies between 0.608% and 5.228%.

These results shows that applying FA in the context of smart mobility is capable of producing satisfying results for our problem. We also point that the obtained results are compounded by the fact that the FA founds its results in a small computational time (3.555 s). In fact and by performing a rather simple and straightforward rules, the FA is capable of performing an intensive global search over the search space. Also, we should note that the performance of our proposed method against the proposed method of Mrad et al. [11] shows an enhance of performance by 3.28%.

Table 1. Results of the FA

Instance size	Average gap %	Average time sec
10	1.105	1.421
15	0.799	1.509
20	0.608	1.852
25	0.964	2.005
30	0.998	2.221
35	1.702	2.496
40	1.759	2.622
45	1.718	2.662
50	1.964	2.878
55	3.123	3.446
60	2.606	3.705
65	2.932	3.918
70	4.119	4.105
75	3.530	4.618
80	4.154	4.754
85	4.332	5.119
90	3.713	5.636
95	4.442	6.554
100	5.228	6.020
Average	2.621	3.555

5 Conclusions and Perspectives

In order to enhance the smart cities performance, we focused in this paper on enhancing the performance of urban transportation tool (smart mobility) while reducing the carbon emission (smart environment). We proposed in this work to use PRT vehicles as a smart mobility tool in the context of smart cities. Next, we focused on implementing a FA for solving an optimization problem related to PRT. The problem consisted on satisfying a set of on-demand origin-destination pairs subject to several constraints such as maximum allowable distance constraint and time window constraint. As the FA is implemented for solving continuous optimization problems, we focused in this paper on the discretization of FA in order to make it able to solve the proposed smart mobility problem. We reported in this paper extensive computational tests on a set of carefully generated instances taken from the literature. The proposed algorithm was shown to get good quality results. As a future work, we are working on the concept of Physical Internet [10] in order to enable hyper-connected transportation service in a sustainable way.

Acknowledgement. The authors would like to thank Dr. Hadhemi Achour for her valuable comments and suggestions to improve the quality of the paper.

References

1. Almoustafa, S., Hanafi, S., Mladenović, N.: New exact method for large asymmetric distance-constrained vehicle routing problem. Eur. J. Oper. Res. **226**(3), 386–394 (2013)
2. Bellman, R.: Dynamic programming and lagrange multipliers. Proc. Natl. Acad. Sci. **42**(10), 767–769 (1956)
3. Chebbi, O., Chaouachi, J.: Reducing the wasted transportation capacity of personal rapid transit systems: an integrated model and multi-objective optimization approach. Transp. Res. Part E: Logistics Transp. Rev. (2015)
4. Fatnassi, E., Chaouachi, J.: A comparison of different lower bounding procedures for the routing of automated guided vehicles in an urban context. Int. J. Appl. Nonlinear Sci. **2**(1–2), 120–135 (2015)
5. Fatnassi, E., Chaouachi, J., Klibi, W.: Planning and operating a shared goods and passengers on-demand rapid transit system for sustainable city-logistics. Transp. Res. Part B: Methodol. **81**, 440–460 (2015)
6. Fatnassi, E., Chebbi, O., Chaouachi, J.: Discrete honeybee mating optimization algorithm for the routing of battery-operated automated guidance electric vehicles in personal rapid transit systems. Swarm Evol. Comput. **26**, 35–49 (2016)
7. Janecki, R., Karoń, G.: Concept of smart cities and economic model of electric buses implementation. In: Mikulski, J. (ed.) TST 2014. CCIS, vol. 471, pp. 100–109. Springer, Heidelberg (2014). doi:10.1007/978-3-662-45317-9_11
8. Karaboga, D., Kaya, E.: An adaptive and hybrid artificial bee colony algorithm (aabc) for anfis training. Appl. Soft Comput. **49**, 423–436 (2016)
9. Laporte, G., Desrochers, M., Nobert, Y.: Two exact algorithms for the distance-constrained vehicle routing problem. Networks **14**(1), 161–172 (1984)
10. Montreuil, B.: Toward a physical internet: meeting the global logistics sustainability grand challenge. Logistics Res. **3**(2–3), 71–87 (2011)
11. Mrad, M., Chebbi, O., Labidi, M., Louly, M.A.: Synchronous routing for personal rapid transit pods. J. Appl. Math. **2014**(1), 1–8 (2014)
12. Mrad, M., Hidri, L.: Optimal consumed electric energy while sequencing vehicle trips in a personal rapid transit transportation system. Comput. Ind. Eng. **79**, 1–9 (2015)
13. Prins, C., Lacomme, P., Prodhon, C.: Order-first split-second methods for vehicle routing problems: a review. Transp. Res. Part C: Emerg. Technol. **40**, 179–200 (2014)
14. Samà, M., Pellegrini, P., D'Ariano, A., Rodriguez, J., Pacciarelli, D.: Ant colony optimization for the real-time train routing selection problem. Transp. Res. Part B: Methodol. **85**, 89–108 (2016)
15. Trilles, S., Calia, A., Belmonte, Ó., Torres-Sospedra, J., Montoliu, R., Huerta, J.: Deployment of an open sensorized platform in a smart city context. Future Gener. Comput. Syst. (2016)
16. Yang, X.S.: Firefly algorithm, stochastic test functions and design optimisation. Int. J. Bio-Inspired Comput. **2**(2), 78–84 (2010)
17. Yang, X.S., Hosseini, S.S.S., Gandomi, A.H.: Firefly algorithm for solving non-convex economic dispatch problems with valve loading effect. Appl. Soft Comput. **12**(3), 1180–1186 (2012)

Hybrid Fuzzy Algorithm for Solving Operational Production Planning Problems

L.A. Gladkov(✉), N.V. Gladkova, and S.A. Gromov

Southern Federal University, Taganrog, Russia
{leo_gladkov, nadyusha.gladkova77}@mail.ru

Abstract. The article deals with the development of methods of solving operational production planning problems. Authors formulated the operational production planning problem statement, determined constraints and the objective function. The scheme of solutions encoding and modified genetic operators are developed to consider the problem character. Authors proposed the hybrid algorithm model based on integration of genetic search methods and fuzzy control approach. Experimental research of developed algorithms characteristics allows us to determine their time complexity. Obtained results show the effectiveness of suggested approach.

Keywords: Operational production planning problems · Genetic algorithm · Fuzzy logic · Scheduling theory · Optimization · Hybrid algorithm

1 Introduction

In accordance with Manufacturing Enterprise Solutions Association International (MESA) definition, operational (elaborate) planning is considered as the process of production schedule drawing up and estimation based on priorities, attributes, characteristics and methods related to a specific character of products and production technology.

Thus, operational planning is reduced to scheduling theory problems [1–4], which requires:

- to appoint an executor for each job;
- to put in order jobs of each executor, i.e. to find their optimal performance sequence to achieve the assigned goal.

In the scheduling theory ordering problems are considered with the requirement that all issues of what and how should be done are resolved. It is suggested that jobs nature do not depend on their performance sequence. Therewith, following assumptions are to use:

1. All assigned jobs are to be performed and to be defined fully. Decomposition of the jobs set into performed and non-performed classes is not included into the ordering problem.
2. Devices allocated to perform jobs are defined uniquely.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_44

3. A set of elementary operations related to each job performance and a set of constraints on the order of their performance are assigned. The manner in which these operations are carried out is defined. It is assumed, that there is at least one device able to perform each operation.

2 Problem Statement

In terms of mechanical engineering the operational production planning problem combines different classes of scheduling theory problems. There is a set of machines (production line) *{M}*, $|M| = m$, each line is characterized by definite list of parameters imposing additional constraints on jobs allocation. The timetable of production lines unavailability determines periods of service or repair works.

- sb_{jl} denotes the beginning of service i on line j;
- se_{jl} denotes the finishing of service i on line j;
- *RQ* denotes total amount of equipment;
- *WT* denotes the matrix of reconfiguration periods required for switching from job i to job j. Each element of the matrix $wt_{ij} \geq 0$.

The finite set of jobs is denoted by *{N},* $|N| = n$, where each job i includes an operation. The job is an elementary problem required to be performed, which is characterized by following parameters:

- number of machine m_i allocated to perform job i, $1 < m_i < m;$
- duration of job performance;
- individual schedule date d_i of job i;
- the criterion of need to use the equipment $rq_i \in \{0, 1\}$;
- the incidence matrix of jobs and production lines R, the matrix element r_{ij} represents the selection priority of the line j to perform the job i,

$$r_{ij} = \begin{cases} r_{ij} \geq 0 \\ r_{ij} = \infty \end{cases},$$

where r_{ij} equals ∞, if the job i is not performed on the line j.

The problem requires finding such decomposition of the jobs set N into m disjoint subsets, which provides following conditions:

1. Distribution of jobs across lines corresponds to the incidence matrix $R, \forall m \in M, i \in N_m \Rightarrow r_{im} > 0$;
2. The lines to perform the jobs is selected with the use of the least priority value

$$\sum_{i \in N} r_{im} \rightarrow min$$

3. There is such schedule (ordering) σ_m: $N_m \rightarrow \{0, 1, \ldots, D\}$ for each subset N_m in terms of planning D, that:
   (a) the sequences of jobs performance on the one line are not to recur,

$$\forall n_{i+1} \in N_m \backslash n_i \Rightarrow \sigma_m(n_{i+1}) \neq \sigma_m(n_i)$$

   (b) the timetable of the line m availability is not to be broken,

$$\forall n_i \in N_m \Rightarrow \begin{cases} \sigma_m(n_i) \notin [sb_{ml}; se_{ml}] \\ \sigma_m(n_i) + t_i \notin [sb_{ml}; se_{ml}] \end{cases}$$

   (c) the number of simultaneously loaded lines is not to be exceeded,

$$\forall i \in \{0, 1, \ldots., D\}, |\{n_i \in N : \sum_{m=1}^{|M|} \sigma_m(n_i)/i\}| \leq m_{\max}, m_{\max} \leq m$$

   (d) conditions of lines reconfiguration are to be fulfilled,

$$\forall n_{i+1} \in N_m \backslash n_i, \sigma_m(n_i) < \sigma_m(n_{i+1}) \Rightarrow \sigma_m(n_{i+1}) - \sigma_m(n_i) - t_i \geq wt_{n_i n_{i+1}}$$

   (e) constraints on simultaneous usage of equipment are:

$$\begin{array}{l} \forall i \in \{0, 1, \ldots., D\} : \{\forall n_j \in N : \\ [\sigma_m(n_j); \sigma_m(n_j) + t_j] \subset i \wedge rq_j = 1\} \leq RQ \end{array}.$$

The common criterion for schedule organization is minimization of the objective function $F \rightarrow min$, where F is considered as the penalty function representing the total jobs deviation from individual schedule dates:

$$F = \sum_{i}^{N} |\sigma(n_i) + t_i) - d_i| \rightarrow min.$$

3 The Algorithm Description

In solving practical problems with the use of genetic algorithms following preliminary tasks are to be accomplished:

(1) to select the way of solutions representation;
(2) to develop genetic operators;
(3) to determine rules of solutions survival;
(4) to generate the initial population.

In terms of the stated problem let us to apply the encoding scheme, when each chromosome consists of required solution entirely. One agent (individual) includes encoded information about the whole plan for the planning period. The downside of the scheme is that chromosomes are very long. However, the objective function of each individual represents the common optimization criterion, and each generation includes a certain set of solutions, which provides faster convergence together with genetic operators with the use of diverse genetic material. Thus, the chromosome structure is represented as a set of production jobs for the planning period i.e. the whole operational plan. The chromosome contains a number of genes: $N_h = Nr_{max} A_r$, where Nr_{max} is the maximum number of production jobs for the planning period; A_r is a number of variable attributes of production jobs. The value of $Nr_{max} = M$ is determined by common number of production jobs formed on the basis of the main production schedule rows.

Thus, the obtained chromosome involves M groups of genes, each group determines corresponding production job completely, that is the first job corresponds to the first group, the second job corresponds to the second one, etc.

The gene value determines the value of corresponding attribute:

- The gene value of ‘duration of job performance’ attribute is a number of hours during which the selected line is loaded by the job.
- The gene value of ‘production line number’ attribute is a sequence number of nonzero element in corresponding row of products and production lines incidence matrix R.
- The gene value of ‘production job order’ is a number of an hour, when the production job begins.

It should be mentioned that the identifier of the product is not encoded individually, they are determined strictly, i.e. each job is related to a certain product clearly. In such encoding way relevant information is contained not only in gene values, but in their position in chromosome, too. This minimizes the chromosome length, providing reduction of the search space. As a result, the convergence time (number of generations to be processed for the purpose of convergence), and the time required for a generation processing decrease, too. Let us note some specific characteristics of such individuals’ representation scheme:

- zero value of job performance duration or job beginning date is interpreted as the absence of production job – so in evolution process the genetic algorithm is able to convergence to optimal number of tasks under condition of $M > M_{opt}$;
- values M are artificial constraints imposed on the search space in such solutions representation.

Authors suggest to encode the chromosome in binary form (Fig. 1).

The chromosome length is calculated as follows:

- $L_h = (L_b + L_{rm} + L_{dm}) N$,
- where L_b is a number of bits required to encode any moment of job beginning during planning period with selected accuracy (in terms of the stated problem with the accuracy of an hour);
- L_{rm} is a number of bits required to encode the alternative line for the product of the production job m. The value of this gene determines sequence number of nonzero element in corresponding row of incidence matrix R;

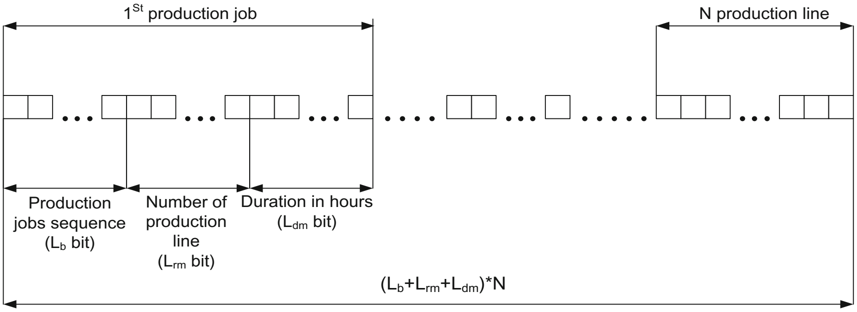

1St production job
N production line
Production jobs sequence (Lb bit)
Number of production line (Lrm bit)
Duration in hours (Ldm bit)
(Lb+Lrm+Ldm)*N

Fig. 1. The chromosome encoding scheme

- L_{dm} is a number of bits required to encode the duration of the job m (in terms of the stated problem the value is multiple of an hour).

Such encoding scheme is sufficiently flexible since it allows us to vary product jobs beginning dates, line numbers and performance durations.

In terms of the stated problem authors suggest to use following genetic operators:

- the selection operator of roulette wheel;
- the reproduction operator. In the process of reproduction which is implemented after selection, chromosomes are copied with the probability proportional to their objective function;
- the crossover operator.

Commonly, the specific characteristic of the stated problem includes penalty functions used in individual's objective function calculation. Besides, it is suggested to modify logic of basic genetic search operators. The modification idea contains the usage of rules applied by subject expert while drawing up the schedule. The essence of these rules is in directed adjustment of production tasks certain parameters to resolve conflicts arising due to violation of constraints conditioned by the problem specific character. Modification assumes changing basic mutation and crossover operators. In particular, there is the specific rule of the allele selection in chromosomes crossover implementation. To visualize modified logic authors show the simplified example considering five jobs and three alternative production lines. Two selected chromosomes represent different solutions that are alternatives of allocation and sequence of jobs performance by production lines. Each rectangle is marked with the index of corresponding job. Rectangles are located horizontally along lines denoting a certain production line. Thus, we obtain the variation of the Gantt diagram commonly used for schedule visualization (Fig. 2).

Jobs are selected for each production line, one from each individual. Genes inherited by a child are determined randomly for each pair (Fig. 3). Herewith, the solution should be tested for duplication. Figure 4 shows the example of jobs allocation during the crossover process.

The result of the crossover operator implementation is shown on Fig. 4. We obtained new solutions, at that excluded incorrect solutions or solutions breaking constraints of jobs sequence on lines.

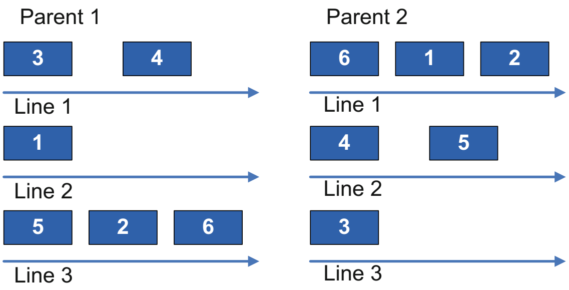
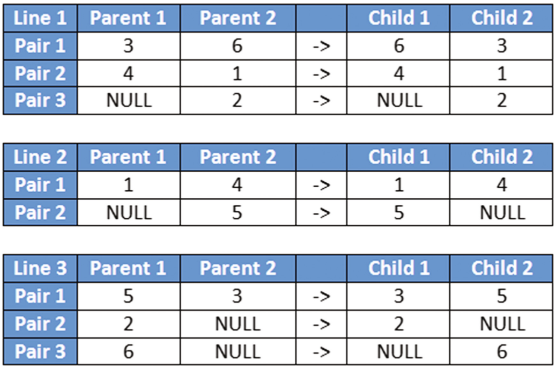


Fig. 2. Alternatives of production planning drawn up with the use of modified crossover operator



Line 1	Parent 1	Parent 2		Child 1	Child 2
Pair 1	3	6	->	6	3
Pair 2	4	1	->	4	1
Pair 3	NULL	2	->	NULL	2

Line 2	Parent 1	Parent 2		Child 1	Child 2
Pair 1	1	4	->	1	4
Pair 2	NULL	5	->	5	NULL

Line 3	Parent 1	Parent 2		Child 1	Child 2
Pair 1	5	3	->	3	5
Pair 2	2	NULL	->	2	NULL
Pair 3	6	NULL	->	NULL	6

Fig. 3. Jobs reallocation resulting from the use of modified crossover operator

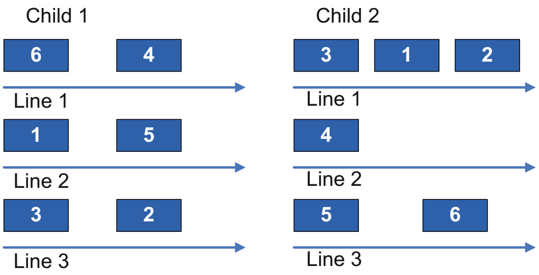


Fig. 4. Results of jobs reallocation with the use of the modified crossover

Thus, in basic crossover operator logic we added the predetermination factor allowing us to exclude potential solutions, i.e. children that do not meet constraints.

- the mutation operator. In terms of the stated problem let us use a multipoint mutation operator. In each chromosome we select N_{OM} pairs of different genes exchanging their values. The value of N_{OM} is calculated as $N_{OM} = \alpha_{OM}L$, where $\alpha_{OM} \in (0;\ 0{,}5)$ is a parameter determining the proportion of gene pairs involved in mutation of the total chromosome length.

The obtained value of N_{OM} is rounded to the nearest larger integer. The value of α_{OM} should be selected from the interval [0,01; 0,03].

- the migration operator is used to exclude premature convergence of the algorithm i.e. local optimum. Let us assume two parallel evolving populations:

$PR^1 = \{pr_1^1, \ldots, pr_i^1, \ldots, pr_{N_{pr}}^1\}$ and $PR^2 = \{pr_1^2, \ldots, pr_i^2, \ldots, pr_{N_{pr}}^2\}$. At a certain stage N_M individuals from the first population move to the second population. The same number of the second population moves to the first one. Thus, population size remains the same. The selection of individuals from both populations is realized on the basis of 'either best or worst' principle. In this case, we select N_M individuals with the best objective function value from the first population and from the second one – individuals with the worst objective function value. Number of individuals involved in the migration is sized by N_M, which value can be calculated as follows: $N_M = \alpha_M N_{pr}$, where $\alpha_M \in (0;\ 1)$ is a parameter determining the proportion of agents (individuals) involved in migration of total population size. The obtained value of N_M is rounded to the nearest larger integer. The value of α_M should be selected from [0,1; 0,3].

The objective function calculation contains following steps:

- to calculate the criterion F;
- to calculate corrections considering constraints;
- to calculate the objective function on the basis of values of the criterion F and corrections.

The criterion F is determined by the function of jobs number that breaks directive requirements during the whole planning period. One of the main factors of effective optimization search is the objective function sensibility. Even little adjustment of the criterion F is to result to the maximum change in relation to other individuals objective function values. The objective function sensibility to solutions varieties is directly connected to the effectiveness while estimating the importance of optimization constraints. Artificial constraints imposed on feasible region apart from encoding scheme strongly restrict the GA search possibilities, due to the creation of artificial local optimums on borders of feasible region formed by reimposed constraints.

The essence of GA corrections connected to penalties imposing assumes the reduction of individual objective function value if the solution represented by this individual overruns feasible region. It should be mentioned that constraints determining feasible region are related to a certain production job or to the whole schedule. They can be subdivided into two groups: jobs penalties or whole schedule penalties. Constraints related to schedule penalties are:

- constraints of technical equipment usage;
- constraints of lines number working a day;
- non-strict constraints of the least priority lines usage.

Penalties for breaking these constraints are calculated clearly in a form of criterion value correction. To calculate the correction let us use following coefficients:

- The coefficient of technical equipment overuse:

$$\alpha_{st} = \sum_{t=1}^{T} \frac{bs(t) - bs_{lim} + |bs(t) - bs_{lim}|}{2} \Bigg/ bs_{lim},$$

where $bs(t)$ is a number of used equipment in a moment t; bs_{lim} is a total number of equipment RQ; T is a planning horizon.

- The coefficient of excess of working lines a day:

$$\alpha_{bq} = \sum_{t=1}^{T} \frac{bq(t) - bq_{\text{lim}} + |bq(t) - bq_{\text{lim}}|}{2} \Bigg/ bq_{\text{lim}}$$

where $bq(t)$ is a number of used equipment in a moment t; bq_{lim} is a limiting number of lines working simultaneously m_{max}; T is a planning horizon.

- The coefficient considering the usage of the least priority lines usage:

$$\alpha_{M_r} = \sum_{t=0}^{T} \frac{M_r(t)}{M(t)},$$

where $M_r(t)$ is a number of production jobs in a day t assigned to production line with the priority value other than the least priority of alternative lines of corresponding job in accordance with the incidence matrix R. $M(t)$ is a total number of production jobs in a day t with nonzero duration.

Corrections are calculated as follows:

$$\begin{aligned}\Delta_{st} &= \tilde{F}^{*} \alpha_{st} \\ \Delta_{bq} &= \tilde{F}^{*} \alpha_{bq} \\ \Delta_{M} &= \tilde{F}^{*}(1 - \alpha_{M_r})\end{aligned}$$

Thus, the value $\Phi = \tilde{F} - \Delta_{st} - \Delta_{bq} - \Delta_{M_r}$ is represented as the objective function value used for the GA operators work.

The next GA step is the stop criterion checking. Stop criterion can be represented as:

- obtaining assigned number of generations;
- obtaining assigned algorithm execution time;
- remaining relative adjustment of population OF average value during a assigned number of generations.

Also, the common stop criterion is obtaining solution meeting the constraints in the optimization problem statement. The last criterion from the three criteria mentioned above represents the convergence factor in a most relevant way.

This criterion can be represented as follows:

$$\frac{|\varPhi_{ij} - \varPhi_{i(j-\delta)}|}{\varPhi_{i(j-\delta)}} \cdot 100\% \leq \varDelta_{\varPhi},$$

where $\varPhi_{ij}$ is the objective function value of individual i on the iteration j; $\varPhi_{i(j-\delta)}$ is the objective function value of individual i on the iteration $(j-\delta)$; δ is a number of iterations to calculate relative objective function change; $\varDelta_{\varPhi}$ is a threshold value in relation to the objective function change.

Values of δ and $\varDelta_{\varPhi}$ are calculated empirically on the basis of research of obtained solutions quality with a certain values, and acceptable algorithm execution time. Initial values can be selected from intervals: $\delta \in [10; 50]$; $\Delta_{\Phi} \in [0.3\%; 3\%]$.

Under the stop condition the obtained solution is captured. Otherwise, genetic operators mentioned above are to be realized.

4 Integration of the Genetic Algorithm and the Adaptive Search

In recent years integrated and hybrid models involving genetic algorithms are of a great interest. We can distinguish approaches of ‘external’ hybridization, for instance, building hybrids of genetic algorithms and evolutionary strategy and neural network meta models [5–7] and ‘internal’ hybridization, when in the context of evolutionary design algorithms are integrated on the basis of evolutionary models, for instance, Darwin model and Lamarck model [8–10].

In this paper authors suggest an ‘external’ hybridization variant, when hybrid fuzzy genetic algorithm combines approaches of fuzzy logic and genetic search in terms of united optimization process. The algorithm scheme is shown on Fig. 5. The main idea of hybridization involves the mathematical tool of fuzzy logic theory used for encoding, calculation of genetic algorithm optimal parameters, genetic operators probability values, fitness function and stop criterion selection. The suggested algorithm can be applied in terms of parallel computing performed on corresponding resources. Modern processors have multicore architecture, which allows us to carry out parallel and distributed computing [11, 12]. In terms of hybrid algorithm logic solutions might be obtained and exchanged simultaneously during one hybrid algorithm iteration.

To improve the quality of genetic search results authors propose to solve the problem of including expert information in evolution process by building fuzzy logic controller which adjusts evolution process parameters values.

Following parameters are used as input [13]:

$$e_1(t) = \frac{f_{ave}(t) - f_{best}(t)}{f_{ave}(t)};\ e_2(t) = \frac{f_{ave}(t) - f_{best}(t)}{f_{worst}(t) - f_{best}(t)};$$
$$e_3(t) = \frac{f_{best}(t) - f_{best}(t-1)}{f_{best}(t)};\ e_4(t) = \frac{f_{ave}(t) - f_{ave}(t-1)}{f_{ave}(t)},$$

Where t is a time step, $f_{best}(t)$ is the best objective function value on iteration t, $f_{best}(t-1)$ - is the best objective function value on iteration $(t-1)$, $f_{worst}(t)$ is the worst

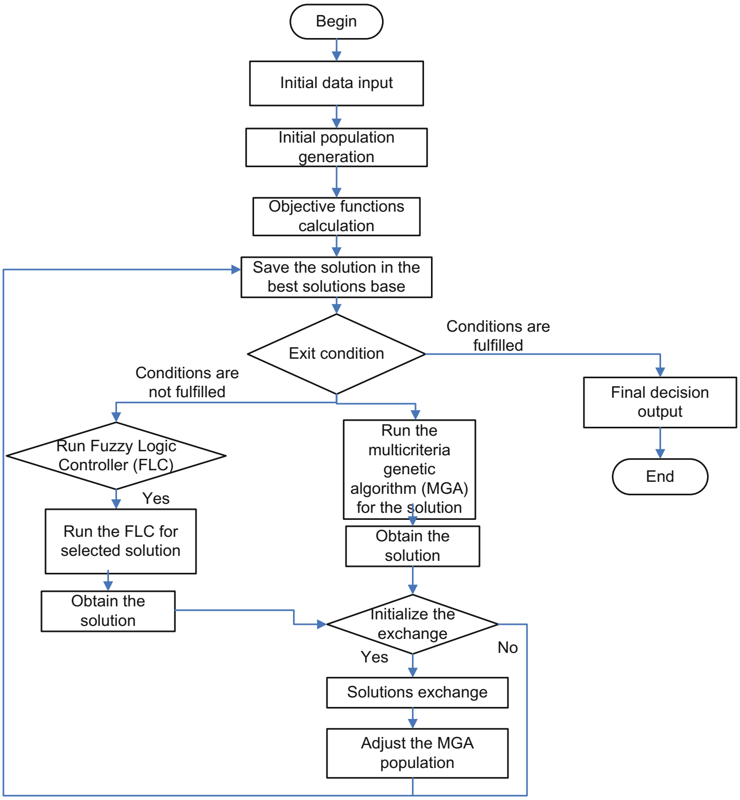

Begin
Initial data input
Initial population generation
Objective functions calculation
Save the solution in the best solutions base
Exit condition
Conditions are fulfilled
Final decision output
End
Conditions are not fulfilled
Run Fuzzy Logic Controller (FLC)
Yes
Run the FLC for selected solution
Obtain the solution
Run the multicriteria genetic algorithm (MGA) for the solution
Obtain the solution
Initialize the exchange
Yes
No
Solutions exchange
Adjust the MGA population

Fig. 5. Hybrid algorithm scheme

objective function value on iteration t, $f_{ave}(t)$ is the average objective function value on iteration t, $f_{ave}(t-1)$ is the average objective function value on iteration $(t-1)$ [14].

Obtained output parameters represent probabilities of crossover, mutation and migration operator.

5 Experimental Research

Let us consider the objective function value as an optimization criterion on the basis of a certain number of algorithm iterations analysis.

The purpose of experiments is to determine the character of problem dimension behavior (values of jobs number N and lines number M), and time spent on solutions search. The convergence is considered as obtaining such objective function D value, that during following δ iterations the change $\varDelta D$ is less than $\varDelta\%$ of previous value.

$$\frac{|D_{i-\delta} - D_i|}{D_{i-\delta}} * 100\% \le \varDelta$$

where $D_{i-\delta}$ is the objective function value on the iteration $(i - \delta)$; D_i is the objective function value on the iteration i; δ is a number of iterations to calculate relative objective function change; i is the current iterations; Δ is the threshold value of relative objective function change.

To carry out the experimental research authors took following values of convergence criterion parameters: δ = 10; Δ = 1%.

The research was carried out on the basis of twelve points selected by experts. Each experiment point of production schedule is determined by initial data vector, which coordinates includes: total amount of jobs N, total amount of lines, number of paired lines $LD = \sum_{j_1=1}^{M} \sum_{j_2=(M-j_1-1)}^{M} ld_{j_1 j_2}$, planning horizon D in hours, number of equipment RQ.

The initial data is shown on the Table 1.

Table 1. Input data for algorithm convergence estimation

№	N	M/LD	D	RQ
1	5	3/2	72	1
2	10	3/2	96	1
3	20	6/2	96	2
4	50	12/4	120	3
5	70	16/6	120	3
6	100	24/8	120	4
7	150	26/8	144	5
8	200	30/10	168	6
9	250	34/10	192	6
10	300	40/12	240	7
11	400	50/16	240	8
12	500	54/18	240	8

For each experiment point authors took common input data:

- Jobs duration are selected randomly from the assigned interval $t_i \in [t_{min}, t_{max}]$;
- A service interval is assigned for the paired line.

As shown on graphics, the convergence time increases linearly at more problem dimension. The adaptive search converge faster that the modified genetic algorithm. Obtained results prove the common assumption that approximate algorithm is relevant for quasioptimal solution search. The behavior character represents almost linear calculation time dependence on the problem dimension, which allows us to assume polynomial time complexity of developed algorithms.

Acknowledgment. This research is supported by the grant from the Russian Foundation for Basic Research (project # 16-01-00715, 17-01-00627).

References

1. Conway, R.M., Maxwell, W.L., Miller, L.W.: Theory of Scheduling, 2nd edn. Dover Publications, Mineola (2004)
2. Pinedo, M.: Scheduling: Theory, Algorithms and Systems, 3rd edn. Springer, New York (2008)
3. Leung, J.Y.T.: Handbook of Scheduling. Chapman & Hall/CRC, Boca Raton (2004)
4. Luger, G.F.: Artificial Intelligence. Structures and Strategies for Complex Problem Solving, 6th edn. Addison Wesley, Boston (2009)
5. Michael, A., Takagi, H.: Dynamic control of genetic algorithms using fuzzy logic techniques. In: Proceedings of the Fifth International Conference on Genetic Algorithms, pp. 76–83. Morgan Kaufmann (1993)
6. Lee, M.A., Takagi, H.: Integrating design stages of fuzzy systems using genetic algorithms. In: Proceedings of the 2nd IEEE International Conference on Fuzzy System, pp. 612–617 (1993)
7. Herrera, F., Lozano, M.: Fuzzy adaptive genetic algorithms: design, taxonomy, and future directions. J. Soft Comput. **7**(8), 545–562 (2003). Springer
8. Gladkov, L.A., Kureichik, V.V., Kureichik, V.M.: Genetic Algorithms. Phizmatlit, Moscow (2010)
9. Gladkov, L.A., Gladkova, N.V., Leiba, S.N.: Hybrid intelligent approach to solving the problem of service data queues. In: Proceeding of 1st International Scientific Conference "Intelligent Information Technologies for Industry" (IITI 2016), vol. 1, pp. 421–433 (2016)
10. Gladkov, L.A., Gladkova, N.V., Legebokov, A.A.: Organization of knowledge management based on hybrid intelligent methods. In: Silhavy, R., Senkerik, R., Oplatkova, Z.K., Prokopova, Z., Silhavy, P. (eds.) Software Engineering in Intelligent Systems. AISC, vol. 349, pp. 107–112. Springer, Cham (2015). doi:10.1007/978-3-319-18473-9_11
11. King, R.T.F.A., Radha, B., Rughooputh, H.C.S.: A fuzzy logic controlled genetic algorithm for optimal electrical distribution network reconfiguration. In: Proceedings of 2004 IEEE International Conference on Networking, Sensing and Control, Taipei, Taiwan, pp. 577–582 (2004)
12. Zhongyang, X., Zhang, Y., Zhang, L., Niu, S.: A parallel classification algorithm based on hybrid genetic algorithm. In: Proceedings of the 6th World Congress on Intelligent Control and Automation, Dalian, China, pp. 3237–3240 (2006)
13. Gladkov, L., Gladkova, N., Leiba, S.: Manufacturing scheduling problem based on fuzzy genetic algorithm. In: Proceeding of IEEE East-West Design and Test Symposium – (EWDTS 2014), Kiev, Ukraine, pp. 209–212 (2014)
14. Gladkov, L.A., Gladkova, N.V., Leiba, S.N.: Electronic computing equipment schemes elements placement based on hybrid intelligence approach. In: Silhavy, R., Senkerik, R., Oplatkova, Z.K., Prokopova, Z., Silhavy, P. (eds.) Intelligent Systems in Cybernetics and Automation Theory. AISC, vol. 348, pp. 35–44. Springer, Cham (2015). doi:10.1007/978-3-319-18503-3_4

Synthesis Production Schedules Based on Ant Colony Optimization Method

Yuriy Skobtsov[1], Olga Chengar[2], Vadim Skobtsov[3](✉), and Alexander N. Pavlov[4,5]

[1] St. Petersburg State National Research Polytechnic University, St. Petersburg, Russia
ya_skobtsov@list.ru

[2] Federal Public Autonomous Educational Institution of the Higher Education Sevastopol State University, Sevastopol, Russian Federation
OVChengar@sevsu.ru

[3] United Institute of Informatics Problems of National Academy of Sciences of Belarus, Minsk, Belarus
vasko_vasko@mail.ru

[4] Saint Petersburg National Research University of Information Technologies, Mechanics and Optics (ITMO), St. Petersburg, Russia
Pavlov62@list.ru

[5] Mozhaisky Military Space Academy, St. Petersburg, Russia

Abstract. It is proposed to use ant algorithms together with the object-oriented simulation models. To optimize the functioning of the automated technological complex machining together with a modified ant algorithm it is designed object model, which allows to calculate the fitness function and evaluate potential solutions. The transition and calculation of the concentration for synthetic pheromone rules are determined for supposed directed ant algorithms.

Keywords: Natural computing · Ant colony algorithm · Production schedules component · Flexible manufacturing systems

1 Introduction

The process of improving the organizational structure of company management strengthens and extends the value of the scope of information technology. The feature of the organization of modern production is the using of flexible manufacturing systems (FMS), allowing to switch from one product to another with minimal time and labor. The use of such systems involves the complex multi-technology tools - flexible production modules (FPM) [1]. They are characterized by complication management and planning process, especially in the small-scale and medium series of multiproduct manufacturing. Therefore, the problem of optimizing production schedules in order to improve performance FMS works always come to the fore in the FMS control process.

To date, we have considerable experience in discrete production planning, which is represented in papers [1, 2]. However, with increasing degree of automation of the industrial enterprises is increasing the need for more powerful systems, short-term

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_45

production planning, especially intra shop planning to compile a detailed schedule download process equipment in a real production environment. This complicates the application of well-known classical methods that explicitly or implicitly assume a variety of facilitation in setting planning problems. An effective means for solving such problems is to use the methods of «Natural computing», which allows to design sub-optimal solutions to problems of real work situations in a short time. Ant algorithms are prominent representative of "natural computing" because are based on behavioral patterns ants as a population of potential solutions and solutions adapted for combinatorial optimization problems, primarily to search optimal paths at graphs [2]. Thus, it can be argued that the ant colony optimization (ACO) method, performing well for solving combinatorial optimization problems, is one of most perspective for planning process of FMS equipment.

2 Statement of the Research Problem

The problem of synthesis for FMS schedules is to allow for an industrial workshop with a given technological equipment to make the procedure for processing details, given the limitations of real work situations in a short time. The model of this process is conveniently represented as a graph, the construction of which is equivalent to the definition of the numbers t_{ij} – time moments of start process for technological operation O_{ij}. The set of numbers $\{t_{ij}\}$ $(i = 1, 2, \ldots n; j = 1, 2, \ldots, m_i)$, that satisfies the formulated conditions, called the schedule of FMS operation, or a graph model $G(i)$.

In addition to the corresponding representation in the form of a graph of the boot process of technological equipment, we must follow the following sequence of actions FMS for the adaptation of ACO method to solve this problem.

1. Design of graph representation for the potential solutions.
2. Determination correction rules for the pheromone concentration that define a positive feedback process.
3. Development of heuristics to determine the arc selection during path design in the graph.
4. Definition of the heuristic ant behavior in the construction solutions in the form of the transition probabilities.
5. Determination of the feasibility of a potential solutions by means of verification procedures, taking into account the problem of limitations.
6. Check the adequacy of the model.

In [3] the problem of the development and construction of graphical-analytical model is investigated in accordance with the selected optimization criterion (1).

$$G = (V, D, P), \tag{1}$$

where V - the set of vertices, each of which represents the position of the processing components; D - set of arcs representing the transition from one process step to another; P – matrix of transfer rules where each arc $(i, j) \in D$ attributed weight P_{ij}.

Based on the developed graphic-analytical model it is proposed the algorithm of ant population evolution for the simulation of the production process in the FMS [3]. This algorithm takes into account the various external influences, such as breaking FMP, planned maintenance works, commissioning of equipment, delays in the supply of materials, etc., as well as the availability of GPS vehicle and warehouse equipment. So when scheduling simulated start and end for each technological operation of the selected type for process equipment.

3 Development of "Directional" Ant Algorithm

The paper goal is design of the "directional" ant algorithm for optimizing production schedules in FMS, which can easily be adapted to the given limitations, taking into account additional problem conditions.

The analysis of existing research in the field of «Natural computing» [4] found that a promising method of solving for complex combinatorial optimization problems is to use ACO method. The advantage of this algorithm for this problem is that this method does not require the construction of a structural model of the production workshop itself, and allows simple modifications that can solve optimization problems of this class with high effectiveness.

Based on the developed graphic-analytical model representations for boot process of the technological equipment FMS [1] the new modification of ant algorithm was developed - "directional" ant algorithm [4, 5]. This modification differs from its analogues by the following features:

1. The proposed method of calculating the probabilities for artificial ant transition at node of the graphical-analytical model based on the analysis of the current situation of production;
2. It is uniquely determined the necessary number of artificial ants in each population, including "elite" individuals, which depends on the repair and readiness for operation of process equipment in the production area;
3. It is defined the available node set for artificial ants to visit next nodes, which contains the top list nodes - the candidates for their visit, and for all ants, besides "elite" ants, the list of forbidden transitions (tabu list) is generated;
4. It is suggested a global rule changes for pheromone concentration on the graph arcs, taking into account not "best", as is commonly believed, [5] but the "worst" paths in the population of artificial ants with a view to improve its during the next iteration;
5. Heuristic information is based on "direct proportional" rule for the transition between nodes.

Let us consider the main features of the proposed method.

1. As previously mentioned for artificial ant the next node selection is not random and made taking into account the current production situation and dynamically changing environment and heuristic information collected to this moment.

   The transition probability of the k-th ant to node O_{ij} determined by the relation (2).

$$\begin{cases} P_{ij,k}(t) = \frac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}(t)]^{\beta}}{\sum\limits_{k=1}^{l} [\tau_{ij}]^{\alpha} [\eta_{ij}(t)]^{\beta}}, & O_{ij} \in N_{ij}^{k} \\ P_{ij,k}(t) = 0, & O_{ij} \notin N_{ij}^{k} \end{cases} \tag{2}$$

where α - significance factor of pheromone concentration;
β - coefficient of importance for heuristic information;
τ_{ij}-the concentration of the pheromone on the arc of the graph;
η_{ij} - heuristic information;
N_{ij}^{k} - list of vertices O_{ij} available for the k-th ant.

2. The preferred choice of next node for the graphical-analytical model based on a "direct proportional" rule of the transition between the nodes, unlike some of the known modifications of ant algorithm [5], when the ant determines the next node, first at random, and then focusing on the amount of pheromone.
"Direct-proportional" transition rule based on heuristic information, which is defined as the ratio (3) of the execution time $T\mathrm{o}_{ij}$ of technological operations to the planned time $T_{S_{ij}}$, which in turn is corrected after each process step of the detail party s, according to the formula (4).

$$\eta_{ij} = \frac{To_{ij}}{Ts_{i}} \tag{3}$$

where $T\mathrm{o}_{ij}$ - the execution time of technological operations of the details party O_{ij} of i-type;
$T_{S_{ij}}$ - term production of a details party of i-type.

$$Ts_{i} = Ts_{i} - Tos_{ij} \tag{4}$$

Selected heuristic calculation formula (3) is determined enough well because, even using the features of the ant algorithm, ant elect not the node, in which he had free after the next manufacturing operation, but the node where the party details will soon end processing. This significantly expands the space search and allows to find a suboptimal solution [4].

3. At each iteration of "directed" ant algorithm all artificial ants incrementally build the paths from beginning to end nodes of the graphical-analytical model. Wherein at each node the artificial ant must select the next node of graph path. If k-th ant is in node O_{ij}, it selects the next node on the basis of the transition probabilities (2).
To calculate the concentration of pheromone in the transition to the next node in the graph ant use "global" rules that promote directed search. These rules make ants move towards the found "worst" solutions with a view to "improving".
This strategy favors exploitation of the search space and is used after the solution is built, that is, after path generation by all ants. Moreover, the pheromone concentration allowed to change only the "worst" (in global sense) ants that built a none optimal path $x(t)$. Thus, for each arc graph pheromone concentration is determined according to the following rule (5 and 6)

$$\tau_{ij}(t+1) = \tau_{ij}(t) + \Delta\tau_{ij}(t), \tag{5}$$

where

$$\Delta\tau_{ij}(t) = \sum_{k=1}^{n_k} \Delta\tau_{ij}^{k}(t). \tag{6}$$

The pheromone concentration, for each artificial ant is computed depending on the desired optimality criterion and calculated by one of the following rules (7–9):

- for maximizing the average load factor of equipment:

$$\Delta\mathrm{\tau}_{\mathrm{ij}}^{\mathrm{k}}(\mathrm{t}) = \frac{1}{\sum\limits_{i,j} Tpr_{ij}^{k}}; \tag{7}$$

  where Tpr_{ij}^{k} - downtime of the k-th equipment before carrying out process step O_{ij}.
- for the problem of minimizing the changeover time of the equipment:

$$\Delta\mathrm{\tau}_{\mathrm{ij}}^{\mathrm{k}}(\mathrm{t}) = \frac{1}{\sum\limits_{\mathrm{i,j}} Tnr_{ij}^{k}}; \tag{8}$$

  where Tnr_{ij}^{k} - setup time to perform FMS manufacturing operation O_{ij}.
- for the problem of minimizing the cycle time manufacturing detail parts:

$$\Delta\mathrm{\tau}_{\mathrm{ij}}^{\mathrm{k}}(\mathrm{t}) = \frac{1}{\sum\limits_{\mathrm{i,j}} Tpr_{ij}^{k} + \mathrm{To}_{\mathrm{ij}}^{\mathrm{k}}}; \tag{9}$$

In (7–9) pheromone deposited is inversely proportional to the quality of the full path to the arcs built ant. When this global information is used to change the concentration of pheromone.

Fumes pheromone will not occur. This is because the need to find more paths which together would provide the best result of the algorithm. On the basis of this feature it is not acceptable that an artificial ant passed the entire path from the first to the last operation. Because when looking for solutions from beginning to end, without taking into account the existence of other ants (production machines), we obtain as a result of the imposition of routes on each other, which is contrary to the mathematical formulation of the synthesis problem schedule. So after selecting the next node artificial ant with a minimum release time will select the arc with a maximum concentration of pheromones. Then pheromone matrix is updated, and the search will begin anew for other ants, until they all move to a new height. This will continue until all the artificial ants (equipment) will not process operations according to the technological production of the map. The method of calculating the amount of pheromone $\Delta\tau_{ij}^{k}$ laid by every ant, prevents premature stagnation.

4. The population of artificial ants always uniquely defined array $K(l)$ (where $l \in [1; nk])])$ and corresponds to the composition of technological equipment used in the production of (FMS and transport units). Thus, the problem of search rational number nk of ants in each population is solved. It should be noted that transport unit is regarded as "elite ant", for which there is no forbidden vertex (tabu list), and which has priority over the other ants in the transition from node to node. This is due to the fact that usually the production workshop it is the busiest transportation process equipment. Thus, a small number of ants convergence algorithm to the shortest path is good enough, while a large number of ants may cause that the search process is not convergent.
5. For all the ants predetermined set of available nodes for visiting vertices that contains the top of the list - the candidates for their visit. And for all the ants, but "elite" it is generated the list of forbidden (tabu list) nodes. The set defines the set of valid nodes for the k-th ant (10). This set may include those vertices Oij transition probability that is not equal to zero, regardless of whether they were visited ant k-m or less (including a shift around the loop). To this end, each ant is created and tracked taboo list. The nodes are removed from the list of N_{ij}^{k} according

$$N_{ij}^{k}(t) = V - \gamma_{ij}^{k}(t), \tag{10}$$

where V - the array of all possible graph nodes,
$v_{ij}^{k}(t)$ the taboo list for the k-th ant.
So for all of artificial ants, except for the "elite" the list of forbidden vertices includes nodes of the graphic analytical model indicating the storage operations for the issuance of blanks, tool and receive the finished product, because they are available to visit only the "elite" ants, i.e., transport equipment.

$$\gamma_{ij}^{k}(t) = \sum_{i=1}^{3}\sum_{j=1}^{n} O_{ij} + \sum_{i=1}^{m}\sum_{j=1}^{n} O_{ij}^{P=0} \tag{11}$$

It should be noted that the list of forbidden nodes (tabu list) for the k-th ant is dynamically updated after each artificial ant transition to a new node, but remains unchanged, only the part of the list that corresponds to the warehouse operations. This modification of the ant colony method includes a local search using the tabu-search for better solutions obtained at each iteration of "directional" ant algorithm.
6. The proposed algorithm uses one of the following convergence criteria [5]:

- end in excess of a user predetermined number of iterations;
- at the end found an acceptable solution;
- end when all the ants follow the same path.

In addition, some improvements have been used from modified ant algorithm of paper [4].

4 Experimental Study of the Parameters

The effectiveness of the ant algorithm depends on a number of control parameters, which include: n_k - number of artificial ants; n_{it} - the maximum number of iterations, τ_0 - the initial concentration of the pheromone, α - the intensification of the pheromone, β - intensification heuristics.

For experimental studies and testing of developed methods and models it is used the technological equipment as the object selected organizational and process technology area of machining parts, such as bodies of rotation.

The criteria of efficiency of functioning of the FMS were investigated as follows:

- Average load factor of technological equipment ($Kzsr \rightarrow max$), because this criterion includes two others: the duration of the production cycle ($T_y \rightarrow min$) and the downtime of production equipment ($T_n \rightarrow min$);
- "just in time" - extreme violation of the terms of manufacturing orders ($T_{av} \rightarrow min$), as this criterion is most relevant in a real production environment.

Values of positive constant α, which determines the effect of the concentration of pheromone, varied in the range of (0.3, 0.5), and factor β, which determines the influence of the heuristic information, varied in the range of (0.5, 0.8). In the tests executed for the selected production section considered different values of the coefficients α and β and their combinations for the two tasks of the proposed performance criteria alone. Figures 1 and 2 shows the production characteristics dependence on the value of positive constant β a value of α equal to 0.3; 0.35 and 0.4 respectively.

Thus, setting the recommended parameters τ_0 control coefficients, α and β, for a variety of performance criteria investigated another important parameter is "directed" the ant algorithm, namely the number of iterations (ant populations) n_t for suboptimal schedules download process equipment. Moreover, experimental studies were carried out taking into account the different number of technological equipment ready for operation (FMS accounted for breakage, scheduled preventive maintenance and a reduction in the number of process equipment in order to save resources).

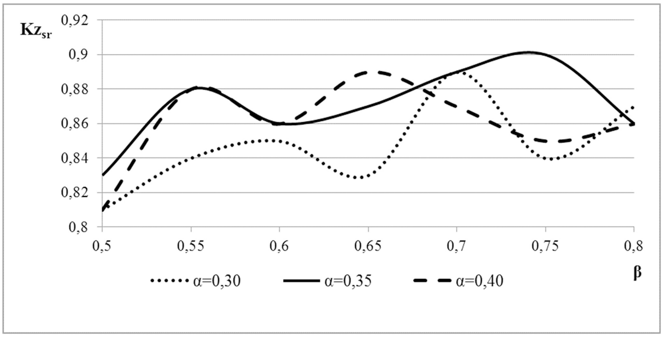


Fig. 1. The dependence of the average load factor of technological equipment from the values of the control coefficients α and β

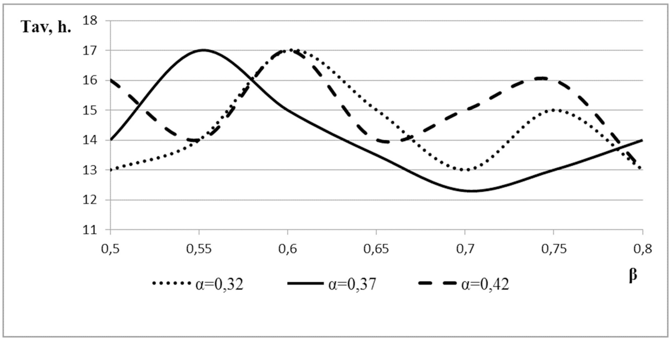


Fig. 2. The dependence of the violation of the order deadlines from the values of the control coefficients α and β

So the situation is viewed from the breakdown of first one ($nk = 6$), and then two FMS ($nk = 5$). The results of these experiments are shown in Figs. 3 and 4, respectively.

Based on the obtained results for the control example, it is possible to conclude that the possibility of reducing the number FMS workshop per unit volume without loss of output, which also corresponds to one of the performance criteria: minimizing the amount used FMS.

Also during the tests "directed" ant algorithm has been found that for the best results of its operation, the coefficients of the algorithm must be experimentally selected for each specific task. Different production sites have their technological features can vary the size of the detail parties and their number, composition and type of equipment - all this affects the result of the algorithm.

Thus, given the specifics of the problem, one can conclude that a single adjustment algorithm parameters (α, β and the number of iterations nt) is not the best option. Also, experiment received several combinations of parameters at the end of the algorithm, it

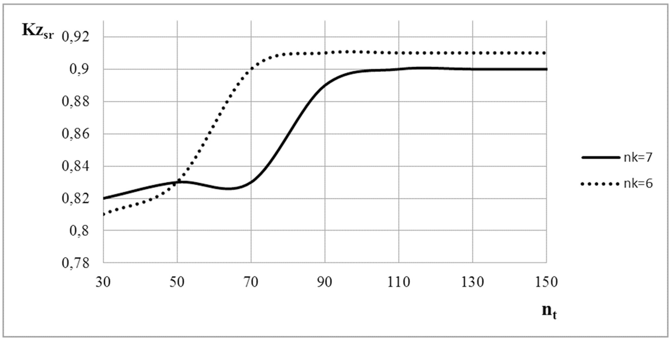


Fig. 3. The dependence of the average load factor of technological equipment (Kzsr) on the number n_t ant populations at varying number of artificial ants

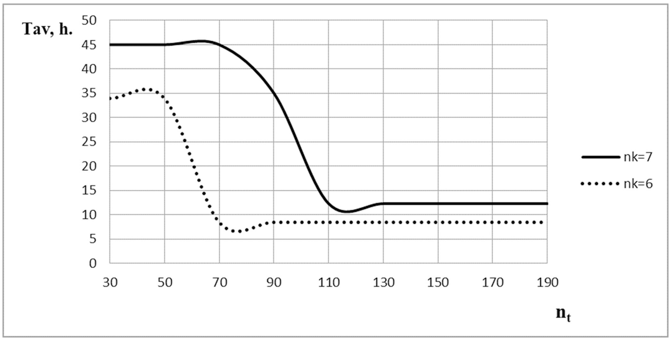


Fig. 4. The dependence of the violation of the order deadlines (Tav) of the number nt ant populations with different numbers nk artificial ants

is necessary to choose the values of the coefficients that allow to find the "best" decision on the selected optimization criterion.

This means that the experimental value of the output parameter data in one production site, can lead to poor results when applying the proposed algorithm to another location. Besides the fact that the optimal values of the "direction" of the coefficients must be obtained experimentally ant algorithm, you must also be able to automatically adjust the settings. In this connection, it was decided to implement a dynamic change in the basic parameters of the algorithm, i.e., after finding the route for all types of processing equipment, to change the coefficients by which it is possible to influence the simulation.

5 Conclusion

In this paper, we propose a modification of the method of ant colonies in order to optimize production schedules in FMS.

1. The proposed method is researched and specified the rules of choice of next node of the graphical-analytical model based on a "directionally proportional to the" rule of the transition between nodes.
2. The "global rules" are proposed for the calculation of the pheromone concentration in the transition ant to the next node in the graph to facilitate search direction.
3. Established and justified the size of the population of artificial ants, the appropriate number of process equipment used in the production of (FMS and transport). Transport unit is regarded as "elite ant.
4. For all the ants predetermined set of available for visiting nodes that contains the list node - the candidates for their visit. And for all the ants, but "elite" defined list of forbidden nodes (tabu list).
5. It is performed numerous computer experiments on test case showed that the effectiveness of the directional ant algorithm increases with the dimension of the problem. It was also found that for the best results of its operation, the coefficients of the algorithm must be experimentally selected for each specific task.

Acknowledgments. The research described in this paper is partially supported by the Russian Foundation for Basic Research (grants **15-07-08391, 15-08-08459, 16-07-00779, 16-08-00510, 16-08-01277, 16-29-09482-ofi-i, 17-08-00797, 17-06-00108, 17-01-00139, 17-20-01214**), grant **074-U01** (ITMO University), project **6.1.1** (Peter the Great St.Petersburg Polytechnic University) supported by Government of Russian Federation, Program STC of Union State "Monitoring-SG" (project **1.4.1-1**, project **6МСГ/13-224-2**), state order of the Ministry of Education and Science of the Russian Federation № 2.3135.2017/K, state research 0073–2014–0009, 0073–2015–0007, International project ERASMUS +, Capacity building in higher education, № 73751-EPP-1-2016-1-DE-EPPKA2-CBHE-JP, Innovative teaching and learning strategies in open modelling and simulation environment for student-centered engineering education.

References

1. Skobtsov, Y., Sekirin, A., Zemlyanskaya, S., Chengar, O., Skobtsov, V., Potryasaev, S.: Application of object-oriented simulation in evolutionary algorithms. In: Silhavy, R., Senkerik, R., Oplatkova, Z.K., Silhavy, P., Prokopova, Z. (eds.) Automation Control Theory Perspectives in Intelligent Systems. AISC, vol. 466, pp. 453–462. Springer, Cham (2016). doi:10.1007/978-3-319-33389-2_43
2. Chengar, O.V.: Development of the "directed" ant algorithm to optimize production schedules. Bull. Kherson Nat. Tech. Univ. **1**(46), 212–217 (2013). (in Russian). ISBN 5-7763-2514-5-Kherson
3. Chengar, O.V.: Graph analytical model of flexible manufacturing modules download automated machine-building enterprise. J. East Ukrainian Nat. Univ. **13**(167), 239–245 (2011). Chenhar, O.V., Savkova, E.O., Lugansk
4. Skobtsov, Y.A., Speransky, D.V.: Evolutionary Computation: Hand Book. The National Open University "INTUIT", Moscow, 331 p. (2015). (in Russian)
5. Dorigo, M.: Swarm intelligence, ant algorithms and ant colony optimization. Reader for CEU Summer University Course «Complex System», pp. 1–34. Central European University, Budapest (2001)

The Use of Deep Learning for Segmentation of Bone Marrow Histological Images

Dorota Oszutowska–Mazurek[1](✉) and Oktawian Knap[2]

[1] Department of Epidemiology and Management, Pomeranian Medical University,
Szczecin, Zolnierska 48 Street, 71-210 Szczecin, Poland
adorotta@op.pl

[2] Department of Forensic Medicine, Pomeranian Medical University, Szczecin,
Powstancow Wielkopolskich 72 Street, 70-111 Szczecin, Poland

Abstract. Proposed solution gives the segmentation of bone marrow histological images, required for further analysis via different methods. Proposed algorithm is based on deep learning using Convolutional Neural Network. More then 50 of ConvNNs where tested with different configurations and learning parameters (learning rate, weight decay). Obtained effectiveness is more then 92%.

Keywords: Bone marrow images · Image segmentation · Deep learning · Convolutional Neural Networks

1 Introduction

Segmentation algorithms are essential for Computer Aided Diagnosis (CAD) of biological specimens. The key problem is the biological variety of images and more complex cases require advanced morphological operations. Some images contain numerous objects of different type and separation is possible with the use of numerous segmentation algorithms. Very low contrast is often observed for some histological images with complex morphological structures.

The example of complex histological image is bone marrow, difficult for segmentation purposes. Segmentation of these images is very important for further analysis that could include quantitative description. There are various factors influencing trabecular structure with impact on histomorphometric assessment. Red bone marrow, essential for hematopoiesis (the production of blood cells) is located in spaces between trabeculae (Fig. 2), in spongy bones (Fig. 1) [2,8,9,17]. There are three types cells in trabecula: osteoblast necessitated for bone synthesis (formation), osteocytes located in lacunae, osteoclasts involved in bone tissue resorption.

The segmentation of the marrow allows further processing of image using CAD tools. Different segmentation algorithms could be proposed for this purpose. One of the most promising techniques are deep Convolutional Neural Networks (ConvNN/CNN) [10,11], because the learning process supports complex image with structures that occur in the image. Such structures, with complex spatial and brightness relations are very often not visible directly for human.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_46

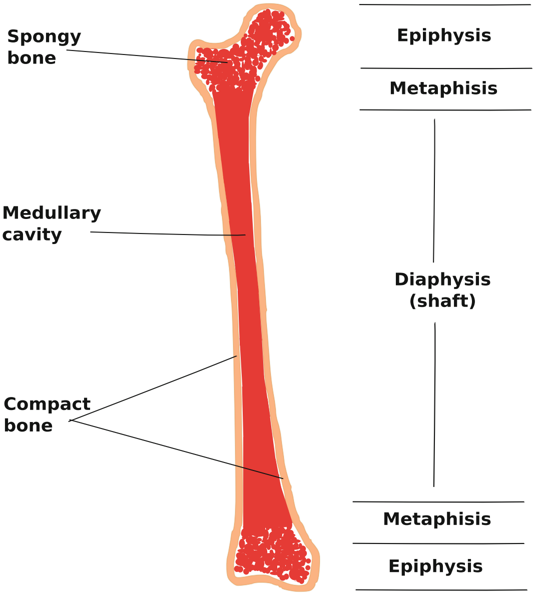


Fig. 1. Bone structure (based on Wikipedia)

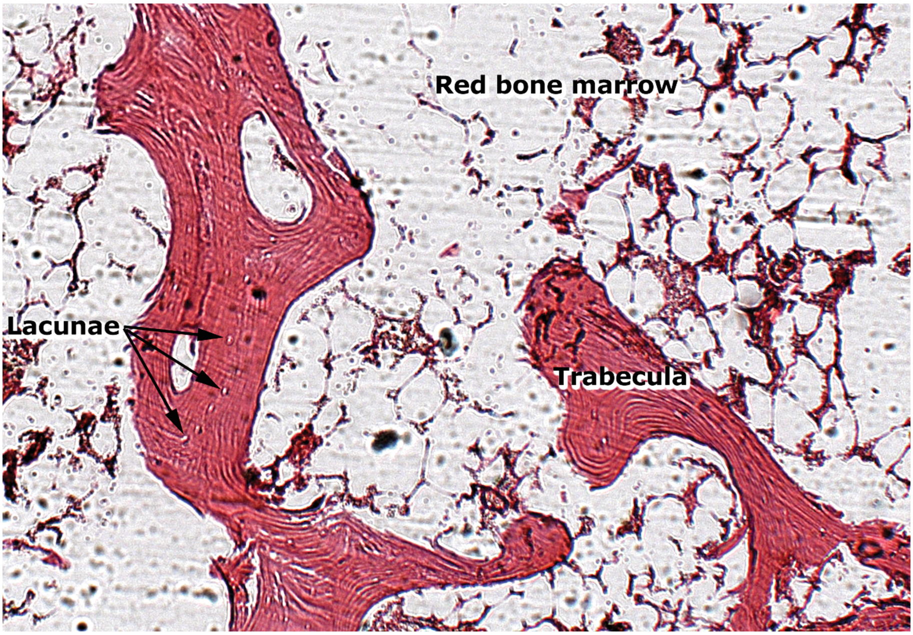


Fig. 2. Example of histological slide of bone marrow

Histological sections of femoral heads were derived from patients after hip joint arthroplasty [5,6]. The images were obtained by Hematoxiline–Eosine (H&E) staining to enhance contrast of elements in bone tissue. H&E staining is a method of chemical preliminary segmentation in histology. Two dyes were used in this technique: basic hematoxylin for the detection of cell nuclei (stained in dark blue) and acid eosine for the detection of cytoplasm (stained in pink). Microscopic images were acquired with the use of microscope Imager D1 (Carl Zeiss) and Axio–CamMRc5 camera that supports 2584 × 1936 resolution.

The obtained image from digital camera is colour, but for the processing in this paper grayscale conversion is applied. The segmentation using color is probably simpler, but morphologically important trabecular structures are visible in grayscale, so it is expected that it is possible to design segmentation algorithm for grayscale images also.

1.1 Related Works

Deep Convolutional Neural Networks are applied for segmentation and classification purposes [1,3,7,12–16] in histopathology and cytopathology. Segmentation is considered even more challenging task than classification [3]. Deep convolutional neural networks with Multiple Instance Learning (MIL) were used for segmentation and classification of microscopy images in [7]. Deep learning was proposed by [3] for the realisation tasks like detection and counting (mitotic events), nuclei segmentation and tissue classification (cancer detection). Prostate cancer identification in biopsy specimens and breast cancer metastasis detection in sentinel lymph nodes were shown in [13] as possible to detect by this new methodology to reduce the workload for pathologists [12] proposed hand–crafted features and convolutional neural networks for gland segmentation in colon histology images. Therefore Deep Convolutional Neural Networks appear as the future, promising method in histopathology.

2 Proposed Segmentation Method

Convolutional Neural Networks use concepts of learning similar to conventional Neural Networks. The main enhancement is the application of convolution (using kernels) for processing data between layers. There are layers dedicated to convolution and the number of weight is reduced because the same weights are shared for calculation of the output of the pixel. Multiple kernels are trained in particular convolutional layer so multiple images are in the output. It provides to significant multiplication of incoming data so another layer typical for ConvNN is used. There are a few types of pooling that are responsible for data reduction (image scaling to lower resolution, using mean or maximal value for local groups of pixels). ConvNN contain a few convolutional layers with pooling but the output of pooling is processed with the use of nonlinear activation function. Is important because convolution and mean pooling provide linear operation and the multiplicity of such layers could be reduced to single linear (single convolutional) layer. The adding of nonlinearity gives the possibility of complex input–to–output transformation support. One of the nonlinear activation function is the ReLU (Rectified Linear Unit) that is very useful, because direct function as well as his derivative could be implemented directly in code without special calculations. Values below zero are transformed to zero, and positive values are passed directly. Such behavior is important for preservation of positive fitting of the kernel to particular image and suppression of negatively correlated image (negative).

The structure of the ConvNN is multilayer, so such processing scheme is assigned to deep learning techniques. Typical last layers of ConvNN are classical neural networks with full connections. The concept of ConvNN is based on the processing image starting from detection of details, to the more complex structures.

ConvNN could be learned with the use of supervised or non–supervised approaches. Typical ConvNN are feedforward networks, but networks with feedback are possible also. The training of ConvNN could be based on the application of learning algorithm to groups of layer or for entire network (all layers together). The training pairs are selected randomly from large database because there are a lot of weights in such network. It is worthy to note that the number of weights is much smaller in comparison to the full connections with independent weights in conventional NN. The number of weights and layers is not optimal for particular problem typically in comparison to the conventional NN. The main advantage of ConvNN is faster learning time, because conventional NN have complex relations between weights and required learning task.

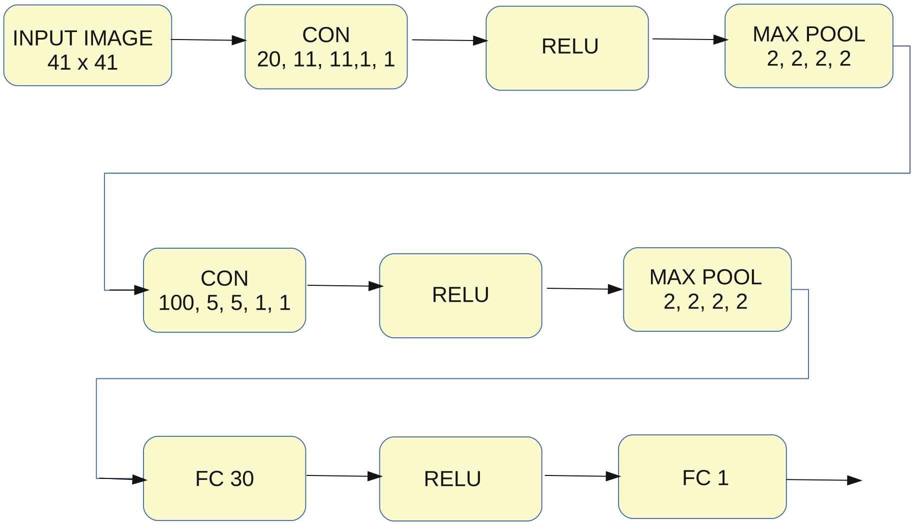


Fig. 3. Structure of ConvNN applied for segmentation

The output of the ConvNN for image segmentation task could binary or 1 from N, where N is the number of classes typically. In this paper binary output is assumed - the presence or absence of trabecula. Multiple output system could be proposed if more features from image should be obtained, e.g. lacunae in trabecula could be another class.

The structure of ConvNN is depicted in Fig. 3.

3 Results

In this work open source machine learning library: dlib is used [4], together with cuDNN for NVidia. GeForce GTX TITAN X (Maxwell) with 3072 CUDA cores, 12 GB of GPU memory and 384–bit memory interface are used. CPU of the computer is used for the augmentation process (rotation, flipping, adding noise) of input image of training pair. The cuDNN is dedicated library for processing data

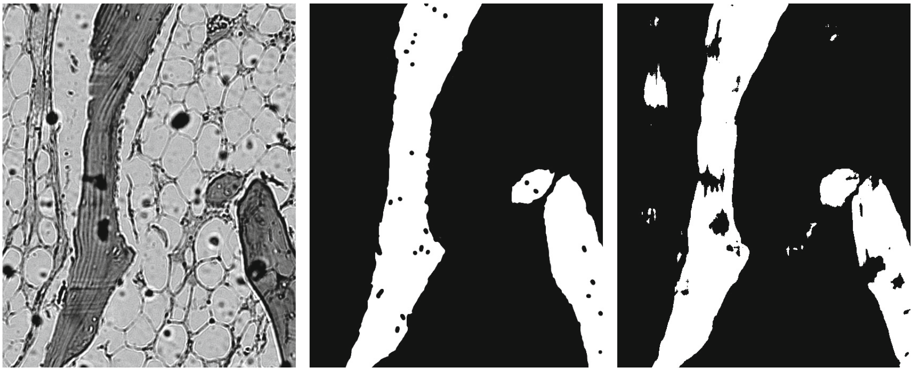


Fig. 4. Example results: original grayscale image (left), reference (middle), obtained segmentation (right)

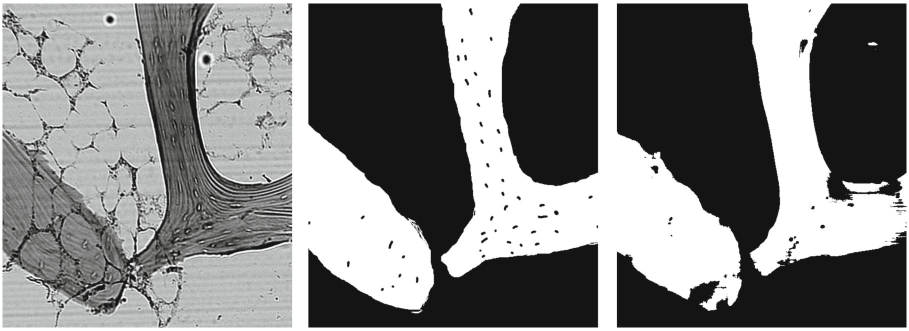


Fig. 5. Example results: original grayscale image (left), reference (middle), obtained segmentation (right)

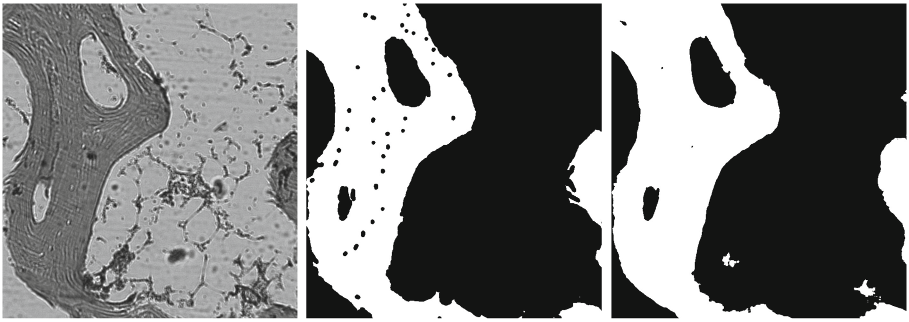


Fig. 6. Example results: original grayscale image (left), reference (middle), obtained segmentation (right)

in ConvNN and is optimized by GPU manufacturer. Dlib allows the configuration of network and the augmentation is processed by the dlib functions also.

The example segmentation of selected images are shown in Figs. 4, 5 and 6. These examples are selected from much larger images. One of the most important problem is the correct creation of mask for learning pair, because some areas are selected as object of interest, but there are background related area that could be trabecula with some probability.

4 Discussion

The obtained effectiveness of segmentation is more then 92%, so up to 8% of pixels are assigned incorrectly as background or trabeculae. There are some areas that are not segmented correctly, that is visible in Figs. 4, 5 and 6. The morphological filtering could improve effectiveness because trabeculae are large areas and most areas selected incorrectly as trabecula are small. The calculation of areas recognized as trabecula and further discrimination depending on size should improve performance. The trabeculae are in most cases surfaces with smooth contour. This property could be also applied for the improving of segmentation, using contour filter.

The lacunae in trabelculae in most cases were incorrectly assigned to trabelcula,

It is also visible that some background artifacts are considered as trabecula element and further improvement of method is required in future work.

The presented results are related to the selected images. More then 50 of ConvNNs where tested with different configurations and learning parameters (learning rate, weight decay). The presented results are for the best configuration Fig. 3. In most cases the effectiveness was greater then 90% and the shortness learning time period was two hours. The longest learning time was about one day.

One of the most important problem was the utilisation of GPU, because for some ConvNN the value was low in 20%–30% range.

5 Conclusions

The configuration of layer in ConvNN could be optimized at meta level using non–gradient optimization technique (e.g. genetic algorithms), but it is difficult to use this technique nowadays, because overall learning process (single learning of ConvNN) is time consuming (hours or days). In most cases it is test–and–try technique, used multiple times. Some observations for the loss values give information about learning progress, but the problem is nonlinear, so learning rate is important factor, but not best one.

The number of learning pairs could be extended with the use of data augmentation (rotation, flipping, adding noise, etc.). This method of augmentation gives more efficient results using smaller dataset, because most training cases are generated from real data. The preparation of learning dataset is the main challenge of creation of such system. Files with histological slides are extremely large (tens or hundreds of thousand of pixels in one direction) so usually data set are large with similar features. Learning algorithms expected large diversity for proper learning process of classification but obtaining of such data is time consuming process because some cases are rare. One of the interesting phenomena is the fact that the number of positive cases (where some medical features related to disease occurs) reduces in time, due to improved detection of particular disease and increased detection of disease in early stages. The improvement of health programs reduces availability of positive cases, so design of CAD systems is difficult.

Acknowledgments. This work is supported by the UE EFRR ZPORR project Z/2.32/I/1.3.1/267 /05 “Szczecin University of Technology – Research and Education Center of Modern Multimedia Technologies” (Poland). We gratefully acknowledge the support of West–Pomeranian University of Technology, Szczecin (Department of Signal Processing and Multimedia Engineering).

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X GPU used for this research.

References

1. Bram, R.: Deep learning in histopathology. Technical report, VU University Amsterdam (2016)
2. Domagala, W., Chosia, M., Urasinska, E.: Atlas of histopathology. Wydawnictwo Lekarskie PZWL (2007)
3. Janowczyk, A., Madabhushi, A.: Deep learning for digital pathology image analysis: a comprehensive tutorial with selected use cases. J. Pathol. Inf. **7**(1), 29 (2016)
4. King, D.E.: Dlib-ml: a machine learning toolkit. J. Mach. Learn. Res. **10**, 1755–1758 (2009)
5. Knapp, O.: Zastosowanie cyfrowej analizy obrazu do ilosciowej oceny histomorfometrycznej preparatow mikroskopowych. Ph.D. thesis, Szczecin (2009)
6. Knapp, O., Waloszczyk, P.: Ilosciowy opis preparatow histopatologicznych glow kosci udowych, w korelacji z wiekiem, przy zastosowaniu cyfrowego analizatora obrazu. Annales Academiae Medicae Stetinensis (2007)

7. Kraus, O.Z., Ba, J.L., Frey, B.J.: Classifying and segmenting microscopy images with deep multiple instance learning. Bioinformatics **32**(12), i52 (2016)
8. Kumar, B., Abbas, A.K., Aster, J.: Robbins Basic Pathology. Elsevier, Philadelphia (2013)
9. Kumar, V., Abbas, A.K., Aster, J.C.: Robbins & Cotran Pathologic Basis of Disease. Elsevier, Philadelphia (2015)
10. LeCun, Y., Kavukcuoglu, K., Farabet, C.: Convolutional networks and applications in vision. In: Proceedings of 2010 IEEE International Symposium on Circuits and Systems, pp. 253–256, May 2010
11. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)
12. Li, W., Manivannan, S., Akbar, S., Zhang, J., Trucco, E., McKenna, S.J.: Gland segmentation in colon histology images using hand-crafted features and convolutional neural networks. In: 2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI), pp. 1405–1408, April 2016
13. Litjens, G., Snchez, C.I., Timofeeva, N., Hermsen, M., Nagtegaal, I., Kovacs, I., Hulsbergen van de Kaa, C., Bult, P., van Ginneken, B., van der Laak, J.: Deep learning as a tool for increased accuracy and efficiency of histopathological diagnosis. Sci. Rep. **6**, 26286 (2016)
14. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, 7–12 June 2015, pp. 3431–3440 (2015)
15. Malon, C., Miller, M., Burger, H.C., Cosatto, E., Graf, H.P.: Identifying histological elements with convolutional neural networks. In: Proceedings of the 5th International Conference on Soft Computing as Transdisciplinary Science and Technology, CSTST 2008, pp. 450–456. ACM, New York (2008)
16. Sirinukunwattana, K., Raza, S.E.A., Tsang, Y.W., Snead, D.R.J., Cree, I.A., Rajpoot, N.M.: Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images. IEEE Trans. Med. Imaging **35**(5), 1196–1206 (2016)
17. Sobotta, J.: Histology. Urban & Schwarzenberg (1983)

Innovative Intelligent Interaction Systems of Loader Cranes and Their Human Operators

Maciej Majewski(✉) and Wojciech Kacalak

Faculty of Mechanical Engineering, Koszalin University of Technology,
Raclawicka 15-17, 75-620 Koszalin, Poland
{maciej.majewski,wojciech.kacalak}@tu.koszalin.pl

Abstract. An overview of the research on the design of interaction systems between lifting devices and their human operators equipped with a speech interface in a natural language, is presented, covering also the integration with augmented reality and interactive manipulators with force feedback. An innovative processing scheme, using several functional modules employing developed hybrid methods, for the interaction has been presented. The proposed design is based on an experimental implementation of the interactive speech-based complex system for controlling loader cranes. It features intelligent methods for meaning analysis of spoken natural languages, analysis and evaluation of command effects, assessment of command safety, and supporting decision-making processes.

Keywords: Interactive system · Intelligent control · Intelligent interface · Speech communication · Natural language processing · Neural networks · Innovative Interfaces · Lifting devices

1 The Design of Intelligent Interaction Systems

Substantial efforts have been put toward the development of intelligent and natural interfaces between humans and machines. Recent research has led to making significant improvements in new developments and new patents in the field of knowledge of spoken natural language processing. Recent advances in development of prototypes of speech-based interactive systems are described in articles in [1–4]. Innovative interaction systems feature speech-based interactive communication [5–7], machine vision - vision systems [8], augmented reality [9] as well as interactive controllers [10] providing force feedback.

The presented research involves the development of a system for controlling a loader crane, equipped with vision and sensorial systems, interactive manipulators with force feedback, as well as a system for bi-directional communication through speech and natural language between the operator and the controlled lifting device. The proposed speech communication system is presented in abbreviated form in Fig. 1. The system is considered intelligent, because it is capable of learning from previous commands to reduce human errors. It is very significant for the development of new effective and flexible methods of precise positioning

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_47

of objects and cargo manipulation. The proposed concept specifies integration of a system for natural-language communication with visual and sensorial systems.

The importance of safety in working with lifting devices is absolutely important. The proposed approach to interaction systems has proven crucial and very useful in safety systems. The safety of the interaction system includes analysis of crane stability for various load conditions and trajectories of load translocation [11]. During the operation cycle, the analysis of mobile crane load handling system stability for selected configurations and operating conditions is performed. It is an important task, because a failure to consider stability conditions in dynamics of the real crane arrangement may lead to loss of stability. As the results of the analysis, variations in stability conditions depending on angular position of the turning column with booms and telescopic booms, positions of telescopic booms, values of boom angle of elevation, load-bearing system components masses as well as on crane loading are processed by the safety system. The safety of the interaction system also includes analysis of the support system reactions for the assessment of mobile crane stability [12]. The developed module is used for computing the ground reaction forces of the crane support system in the entire operating range. The verification of the mathematical model was performed using the finite element method. The results of numerical computations were used to analyze the stability of the mobile crane load handling system for selected configurations and operating conditions. The results of the simulation provide changes of the reaction forces in the support system and the envelope of the load path for given load capacities and reach of the crane.

The proposed interactive system (Fig. 2) contains many specialized modules and it is divided into the following subsystems: a subsystem for voice communication between a human operator and the mobile crane, a subsystem for natural language meaning analysis, a subsystem for operator's command effect analysis and evaluation, a subsystem for command safety assessment, a subsystem for command execution, a subsystem of supervision and diagnostics, a subsystem of decision-making and learning, a subsystem of interactive manipulators with force feedback, and a visual and sensorial subsystem. The novelty of the system also consists of inclusion of several adaptive layers in the spoken natural language command interface for human biometric identification, speech recognition, word recognition, sentence syntax and segment analysis, command analysis and recognition, command effect analysis and safety assessment, process supervision and human reaction assessment.

The subsystem of visually aiding loader crane control with augmented reality generates virtual images of augmented reality (including markers, points), and also projects images from the vision system on a monitor setup or inside of 3D-vision goggles. The operator's extended field of view contains a camera system in configurations: parallactic setup - synchronization with the operator's head, and a system of stationary cameras making a virtual camera.

The subsystem for speech communication is used to perform the following tasks: processing the operator's spoken commands, operator biometric identification, converting voice commands to text and numerical notation, handling

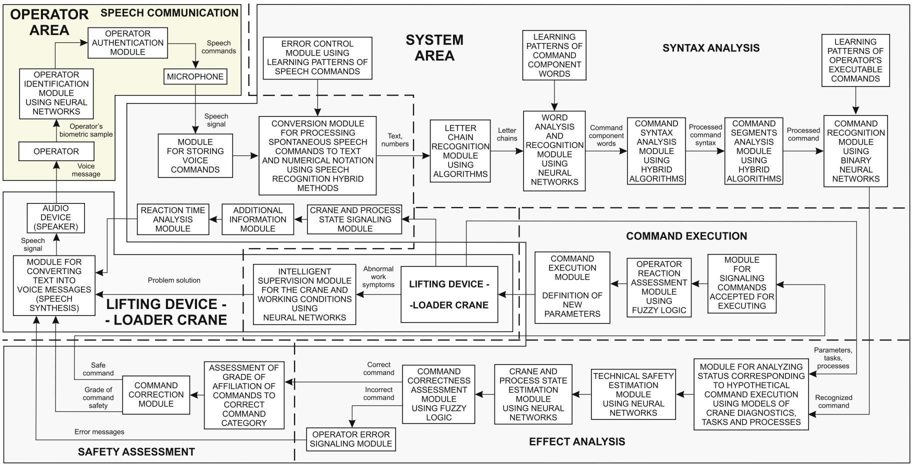

OPERATOR AREA
SPEECH COMMUNICATION
OPERATOR AUTHENTICATION MODULE
Speech commands
OPERATOR IDENTIFICATION MODULE USING NEURAL NETWORKS
MICROPHONE
Operator's biometric sample
OPERATOR
Voice message
ERROR CONTROL MODULE USING LEARNING PATTERNS OF SPEECH COMMANDS
SYSTEM AREA
LEARNING PATTERNS OF COMMAND COMPONENT WORDS
SYNTAX ANALYSIS
LEARNING PATTERNS OF OPERATOR'S EXECUTABLE COMMANDS
Speech signal
MODULE FOR STORING VOICE COMMANDS
CONVERSION MODULE FOR PROCESSING SPONTANEOUS SPEECH COMMANDS TO TEXT AND NUMERICAL NOTATION USING SPEECH RECOGNITION HYBRID METHODS
Text, numbers
LETTER CHAIN RECOGNITION MODULE USING ALGORITHMS
Letter chains
WORD ANALYSIS AND RECOGNITION MODULE USING NEURAL NETWORKS
Command component words
COMMAND SYNTAX ANALYSIS MODULE USING HYBRID ALGORITHMS
Processed command syntax
COMMAND SEGMENTS ANALYSIS MODULE USING HYBRID ALGORITHMS
Processed command
COMMAND RECOGNITION MODULE USING BINARY NEURAL NETWORKS
AUDIO DEVICE (SPEAKER)
REACTION TIME ANALYSIS MODULE
ADDITIONAL INFORMATION MODULE
CRANE AND PROCESS STATE SIGNALING MODULE
COMMAND EXECUTION
Speech signal
MODULE FOR CONVERTING TEXT INTO VOICE MESSAGES (SPEECH SYNTHESIS)
Problem solution
LIFTING DEVICE - - LOADER CRANE
INTELLIGENT SUPERVISION MODULE FOR THE CRANE AND WORKING CONDITIONS USING NEURAL NETWORKS
Abnormal work symptoms
LIFTING DEVICE - -LOADER CRANE
COMMAND EXECUTION MODULE DEFINITION OF NEW PARAMETERS
OPERATOR REACTION ASSESSMENT MODULE USING FUZZY LOGIC
MODULE FOR SIGNALING COMMANDS ACCEPTED FOR EXECUTING
Parameters, tasks, processes
Safe command
Grade of command safety
COMMAND CORRECTION MODULE
ASSESSMENT OF GRADE OF AFFILIATION OF COMMANDS TO CORRECT COMMAND CATEGORY
Correct command
Incorrect command
COMMAND CORRECTNESS ASSESSMENT MODULE USING FUZZY LOGIC
CRANE AND PROCESS STATE ESTIMATION MODULE USING NEURAL NETWORKS
TECHNICAL SAFETY ESTIMATION MODULE USING NEURAL NETWORKS
MODULE FOR ANALYZING STATUS CORRESPONDING TO HYPOTHETICAL COMMAND EXECUTION USING MODELS OF CRANE DIAGNOSTICS, TASKS AND PROCESSES
Recognized command
Error messages
OPERATOR ERROR SIGNALING MODULE
SAFETY ASSESSMENT
EFFECT ANALYSIS

Fig. 1. Intelligent speech communication systems for controlling lifting devices.

errors, analysis of words, recognition of words, analysis of commands' syntax, analysis of commands' segments, recognition of commands, meaning analysis of natural-language messages, as well as converting text into voice messages (speech synthesis [13]). The voice communication subsystem also provides voice feedback to the operator including reporting on the crane's working conditions' safety and expert information for exploitation and controlling. It is also communicated with a subsystem of interactive manipulators with force feedback.

The subsystem of effect analysis and evaluation of the operator's commands is designed for the following tasks: analysis of a state after hypothetical execution of a command, evaluation of technical safety, evaluation of crane systems' and process's states, evaluation of crane working conditions' safety, forecasting of process states' causes, evaluation of commands' correctness, as well as detection of the operator's errors. The commands' safety assessment subsystem is assigned to evaluate the command correctness. The subsystem of the command execution is capable of signaling of process states. The execution of commands involves determination of process's parameters and its manner of execution for the configuration of the crane. The subsystem for supervision and diagnostics implements crane diagnostics, supervision of the controlling process, remote supervision with mobile technologies. It also includes the tasks related to measurements of the crane's working space and collection of geometrical data using photogrammetric techniques. The decision-making and learning subsystem is composed of expert systems, and the intelligent learning kernel integrated with augmented reality. The system is also linked with the interactive manipulators providing force feedback, which include the operator's shoulders' movement interactive scanner for gesture-based manipulation, a shifter with the forces-measuring system, and a multi-axis joystick. It is a connection to a force feedback-based communication channel (crane's working conditions diagnostic information) containing force

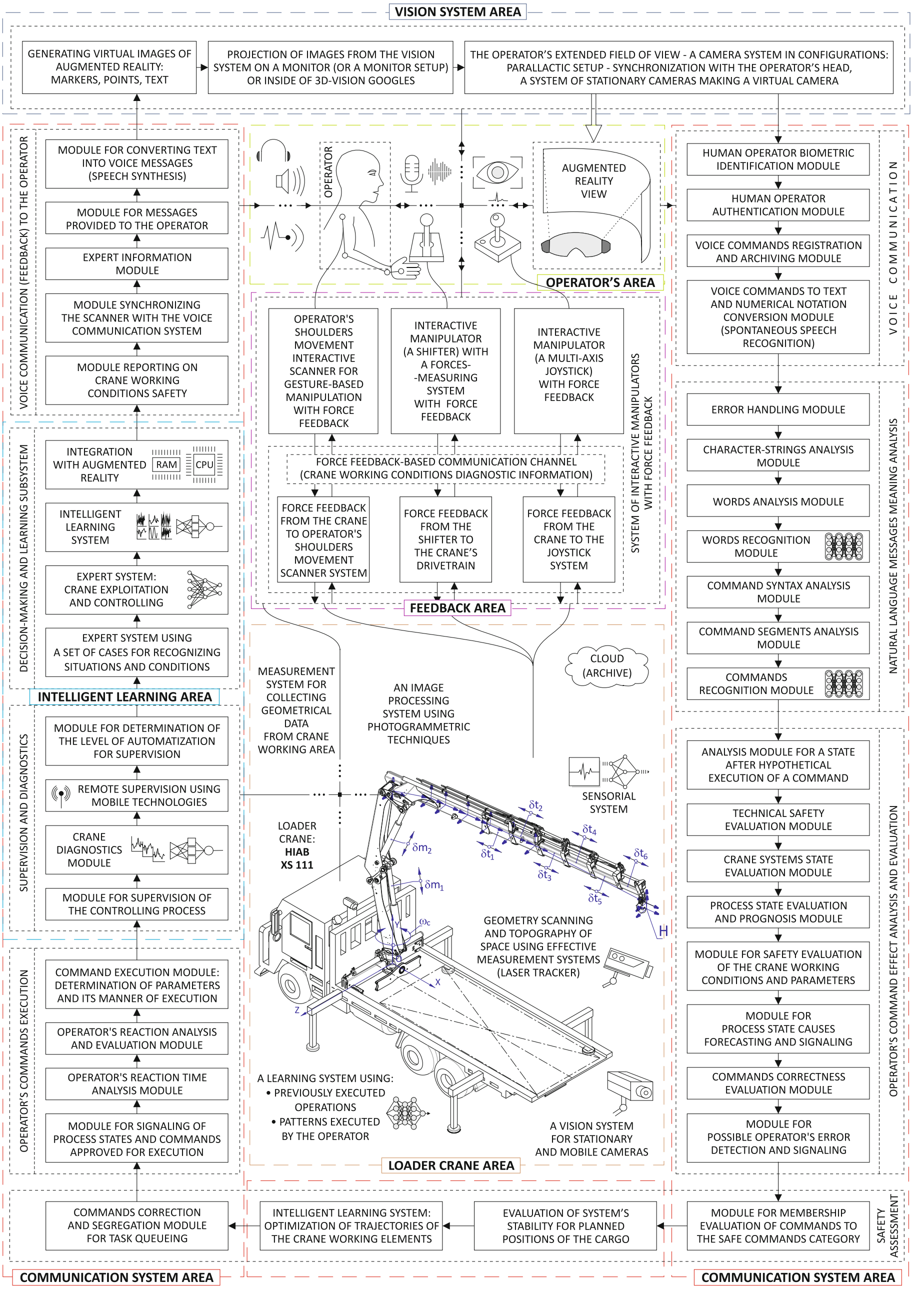

VISION SYSTEM AREA
GENERATING VIRTUAL IMAGES OF AUGMENTED REALITY: MARKERS, POINTS, TEXT
PROJECTION OF IMAGES FROM THE VISION SYSTEM ON A MONITOR (OR A MONITOR SETUP) OR INSIDE OF 3D-VISION GOOGLES
THE OPERATOR'S EXTENDED FIELD OF VIEW - A CAMERA SYSTEM IN CONFIGURATIONS: PARALLACTIC SETUP - SYNCHRONIZATION WITH THE OPERATOR'S HEAD, A SYSTEM OF STATIONARY CAMERAS MAKING A VIRTUAL CAMERA
VOICE COMMUNICATION (FEEDBACK) TO THE OPERATOR
MODULE FOR CONVERTING TEXT INTO VOICE MESSAGES (SPEECH SYNTHESIS)
MODULE FOR MESSAGES PROVIDED TO THE OPERATOR
EXPERT INFORMATION MODULE
MODULE SYNCHRONIZING THE SCANNER WITH THE VOICE COMMUNICATION SYSTEM
MODULE REPORTING ON CRANE WORKING CONDITIONS SAFETY
OPERATOR
AUGMENTED REALITY VIEW
OPERATOR'S AREA
HUMAN OPERATOR BIOMETRIC IDENTIFICATION MODULE
HUMAN OPERATOR AUTHENTICATION MODULE
VOICE COMMANDS REGISTRATION AND ARCHIVING MODULE
VOICE COMMANDS TO TEXT AND NUMERICAL NOTATION CONVERSION MODULE (SPONTANEOUS SPEECH RECOGNITION)
VOICE COMMUNICATION
OPERATOR'S SHOULDERS MOVEMENT INTERACTIVE SCANNER FOR GESTURE-BASED MANIPULATION WITH FORCE FEEDBACK
INTERACTIVE MANIPULATOR (A SHIFTER) WITH A FORCES--MEASURING SYSTEM WITH FORCE FEEDBACK
INTERACTIVE MANIPULATOR (A MULTI-AXIS JOYSTICK) WITH FORCE FEEDBACK
SYSTEM OF INTERACTIVE MANIPULATORS WITH FORCE FEEDBACK
FORCE FEEDBACK-BASED COMMUNICATION CHANNEL (CRANE WORKING CONDITIONS DIAGNOSTIC INFORMATION)
FORCE FEEDBACK FROM THE CRANE TO OPERATOR'S SHOULDERS MOVEMENT SCANNER SYSTEM
FORCE FEEDBACK FROM THE SHIFTER TO THE CRANE'S DRIVETRAIN
FORCE FEEDBACK FROM THE CRANE TO THE JOYSTICK SYSTEM
FEEDBACK AREA
DECISION-MAKING AND LEARNING SUBSYSTEM
INTEGRATION WITH AUGMENTED REALITY
RAM
CPU
INTELLIGENT LEARNING SYSTEM
EXPERT SYSTEM: CRANE EXPLOITATION AND CONTROLLING
EXPERT SYSTEM USING A SET OF CASES FOR RECOGNIZING SITUATIONS AND CONDITIONS
INTELLIGENT LEARNING AREA
ERROR HANDLING MODULE
CHARACTER-STRINGS ANALYSIS MODULE
WORDS ANALYSIS MODULE
WORDS RECOGNITION MODULE
COMMAND SYNTAX ANALYSIS MODULE
COMMAND SEGMENTS ANALYSIS MODULE
COMMANDS RECOGNITION MODULE
NATURAL LANGUAGE MESSAGES MEANING ANALYSIS
MEASUREMENT SYSTEM FOR COLLECTING GEOMETRICAL DATA FROM CRANE WORKING AREA
AN IMAGE PROCESSING SYSTEM USING PHOTOGRAMMETRIC TECHNIQUES
CLOUD (ARCHIVE)
SENSORIAL SYSTEM
SUPERVISION AND DIAGNOSTICS
MODULE FOR DETERMINATION OF THE LEVEL OF AUTOMATIZATION FOR SUPERVISION
REMOTE SUPERVISION USING MOBILE TECHNOLOGIES
CRANE DIAGNOSTICS MODULE
MODULE FOR SUPERVISION OF THE CONTROLLING PROCESS
LOADER CRANE: HIAB XS 111
δt2
δt4
δt6
δt1
δt3
δt5
δm2
δm1
ωc
H
x
y
z
GEOMETRY SCANNING AND TOPOGRAPHY OF SPACE USING EFFECTIVE MEASUREMENT SYSTEMS (LASER TRACKER)
ANALYSIS MODULE FOR A STATE AFTER HYPOTHETICAL EXECUTION OF A COMMAND
TECHNICAL SAFETY EVALUATION MODULE
CRANE SYSTEMS STATE EVALUATION MODULE
PROCESS STATE EVALUATION AND PROGNOSIS MODULE
MODULE FOR SAFETY EVALUATION OF THE CRANE WORKING CONDITIONS AND PARAMETERS
MODULE FOR PROCESS STATE CAUSES FORECASTING AND SIGNALING
COMMANDS CORRECTNESS EVALUATION MODULE
MODULE FOR POSSIBLE OPERATOR'S ERROR DETECTION AND SIGNALING
OPERATOR'S COMMAND EFFECT ANALYSIS AND EVALUATION
OPERATOR'S COMMANDS EXECUTION
COMMAND EXECUTION MODULE: DETERMINATION OF PARAMETERS AND ITS MANNER OF EXECUTION
OPERATOR'S REACTION ANALYSIS AND EVALUATION MODULE
OPERATOR'S REACTION TIME ANALYSIS MODULE
MODULE FOR SIGNALING OF PROCESS STATES AND COMMANDS APPROVED FOR EXECUTION
A LEARNING SYSTEM USING:
• PREVIOUSLY EXECUTED OPERATIONS
• PATTERNS EXECUTED BY THE OPERATOR
A VISION SYSTEM FOR STATIONARY AND MOBILE CAMERAS
LOADER CRANE AREA
COMMANDS CORRECTION AND SEGREGATION MODULE FOR TASK QUEUEING
INTELLIGENT LEARNING SYSTEM: OPTIMIZATION OF TRAJECTORIES OF THE CRANE WORKING ELEMENTS
EVALUATION OF SYSTEM'S STABILITY FOR PLANNED POSITIONS OF THE CARGO
MODULE FOR MEMBERSHIP EVALUATION OF COMMANDS TO THE SAFE COMMANDS CATEGORY
SAFETY ASSESSMENT
COMMUNICATION SYSTEM AREA
COMMUNICATION SYSTEM AREA

Fig. 2. Designed structure of an innovative system for interaction of the loader crane (Hiab XS 111) with its operator equipped with a speech interface, vision and sensorial systems, and interactive manipulators with force feedback.

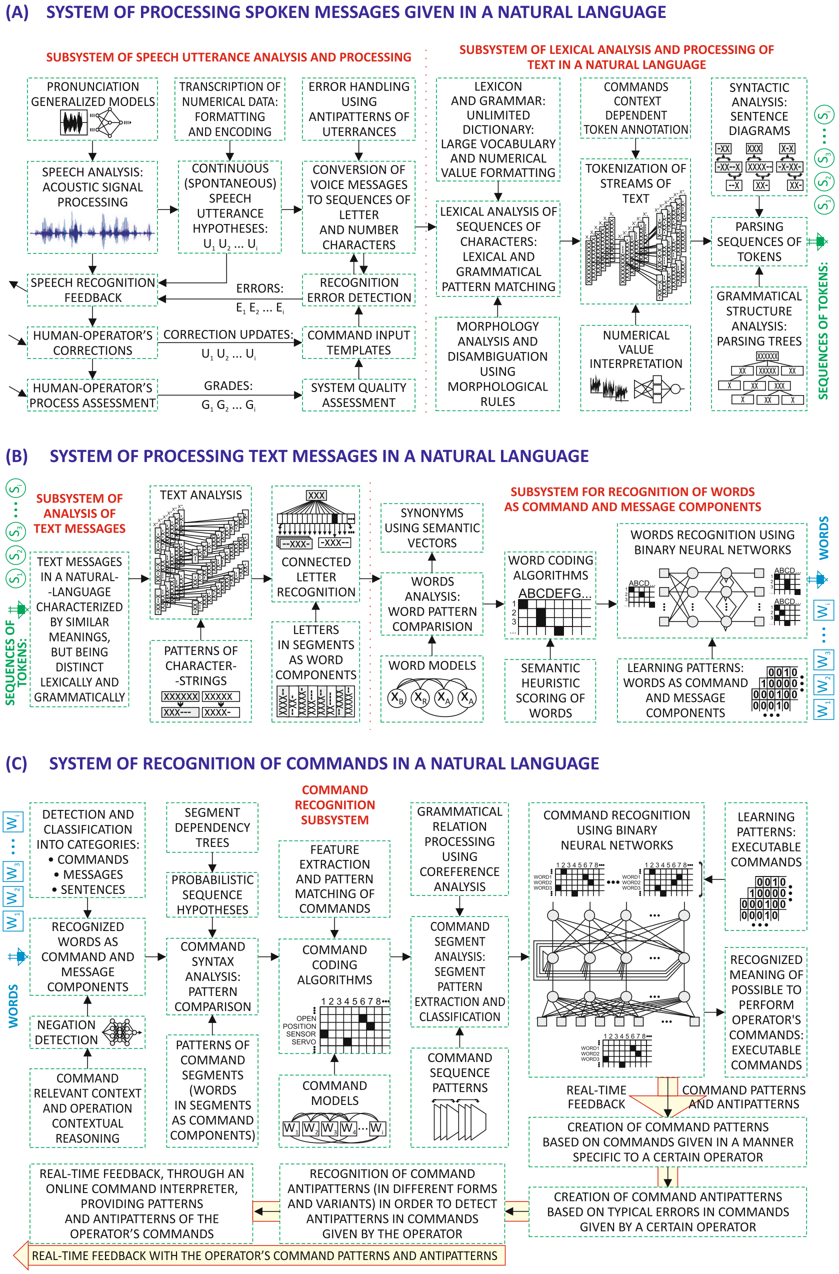

(A) SYSTEM OF PROCESSING SPOKEN MESSAGES GIVEN IN A NATURAL LANGUAGE
SUBSYSTEM OF SPEECH UTTERANCE ANALYSIS AND PROCESSING
SUBSYSTEM OF LEXICAL ANALYSIS AND PROCESSING OF TEXT IN A NATURAL LANGUAGE
PRONUNCIATION GENERALIZED MODELS
TRANSCRIPTION OF NUMERICAL DATA: FORMATTING AND ENCODING
ERROR HANDLING USING ANTIPATTERNS OF UTERRANCES
SPEECH ANALYSIS: ACOUSTIC SIGNAL PROCESSING
CONTINUOUS (SPONTANEOUS) SPEECH UTTERANCE HYPOTHESES: $U_1\ U_2\ ...\ U_i$
CONVERSION OF VOICE MESSAGES TO SEQUENCES OF LETTER AND NUMBER CHARACTERS
SPEECH RECOGNITION FEEDBACK
ERRORS: $E_1\ E_2\ ...\ E_i$
RECOGNITION ERROR DETECTION
HUMAN-OPERATOR'S CORRECTIONS
CORRECTION UPDATES: $U_1\ U_2\ ...\ U_i$
COMMAND INPUT TEMPLATES
HUMAN-OPERATOR'S PROCESS ASSESSMENT
GRADES: $G_1\ G_2\ ...\ G_i$
SYSTEM QUALITY ASSESSMENT
LEXICON AND GRAMMAR: UNLIMITED DICTIONARY: LARGE VOCABULARY AND NUMERICAL VALUE FORMATTING
COMMANDS CONTEXT DEPENDENT TOKEN ANNOTATION
SYNTACTIC ANALYSIS: SENTENCE DIAGRAMS
LEXICAL ANALYSIS OF SEQUENCES OF CHARACTERS: LEXICAL AND GRAMMATICAL PATTERN MATCHING
TOKENIZATION OF STREAMS OF TEXT
PARSING SEQUENCES OF TOKENS
MORPHOLOGY ANALYSIS AND DISAMBIGUATION USING MORPHOLOGICAL RULES
NUMERICAL VALUE INTERPRETATION
GRAMMATICAL STRUCTURE ANALYSIS: PARSING TREES
SEQUENCES OF TOKENS:
S_1 S_2 S_3 ... S_i
(B) SYSTEM OF PROCESSING TEXT MESSAGES IN A NATURAL LANGUAGE
SUBSYSTEM OF ANALYSIS OF TEXT MESSAGES
TEXT MESSAGES IN A NATURAL-LANGUAGE CHARACTERIZED BY SIMILAR MEANINGS, BUT BEING DISTINCT LEXICALLY AND GRAMMATICALLY
TEXT ANALYSIS
PATTERNS OF CHARACTER-STRINGS
CONNECTED LETTER RECOGNITION
LETTERS IN SEGMENTS AS WORD COMPONENTS
SUBSYSTEM FOR RECOGNITION OF WORDS AS COMMAND AND MESSAGE COMPONENTS
SYNONYMS USING SEMANTIC VECTORS
WORDS ANALYSIS: WORD PATTERN COMPARISION
WORD MODELS
WORD CODING ALGORITHMS
ABCDEFG...
SEMANTIC HEURISTIC SCORING OF WORDS
WORDS RECOGNITION USING BINARY NEURAL NETWORKS
LEARNING PATTERNS: WORDS AS COMMAND AND MESSAGE COMPONENTS
WORDS
W_1 W_2 W_3 ... W_i
(C) SYSTEM OF RECOGNITION OF COMMANDS IN A NATURAL LANGUAGE
COMMAND RECOGNITION SUBSYSTEM
DETECTION AND CLASSIFICATION INTO CATEGORIES: • COMMANDS • MESSAGES • SENTENCES
RECOGNIZED WORDS AS COMMAND AND MESSAGE COMPONENTS
NEGATION DETECTION
COMMAND RELEVANT CONTEXT AND OPERATION CONTEXTUAL REASONING
SEGMENT DEPENDENCY TREES
PROBABILISTIC SEQUENCE HYPOTHESES
COMMAND SYNTAX ANALYSIS: PATTERN COMPARISON
PATTERNS OF COMMAND SEGMENTS (WORDS IN SEGMENTS AS COMMAND COMPONENTS)
FEATURE EXTRACTION AND PATTERN MATCHING OF COMMANDS
COMMAND CODING ALGORITHMS
OPEN POSITION SENSOR SERVO
COMMAND MODELS
GRAMMATICAL RELATION PROCESSING USING COREFERENCE ANALYSIS
COMMAND SEGMENT ANALYSIS: SEGMENT PATTERN EXTRACTION AND CLASSIFICATION
COMMAND SEQUENCE PATTERNS
COMMAND RECOGNITION USING BINARY NEURAL NETWORKS
WORD1 WORD2 WORD3
LEARNING PATTERNS: EXECUTABLE COMMANDS
RECOGNIZED MEANING OF POSSIBLE TO PERFORM OPERATOR'S COMMANDS: EXECUTABLE COMMANDS
REAL-TIME FEEDBACK
COMMAND PATTERNS AND ANTIPATTERNS
CREATION OF COMMAND PATTERNS BASED ON COMMANDS GIVEN IN A MANNER SPECIFIC TO A CERTAIN OPERATOR
CREATION OF COMMAND ANTIPATTERNS BASED ON TYPICAL ERRORS IN COMMANDS GIVEN BY A CERTAIN OPERATOR
RECOGNITION OF COMMAND ANTIPATTERNS (IN DIFFERENT FORMS AND VARIANTS) IN ORDER TO DETECT ANTIPATTERNS IN COMMANDS GIVEN BY THE OPERATOR
REAL-TIME FEEDBACK, THROUGH AN ONLINE COMMAND INTERPRETER, PROVIDING PATTERNS AND ANTIPATTERNS OF THE OPERATOR'S COMMANDS
REAL-TIME FEEDBACK WITH THE OPERATOR'S COMMAND PATTERNS AND ANTIPATTERNS

Fig. 3. Recognition of speech commands in a natural language using patterns and antipatterns of commands, consisting of: (A) a system of processing spoken messages given in a natural language, (B) a system of processing text messages in a natural language, (C) a system of recognition of commands in a natural language.

feedback from the crane to the operator's shoulders' movement scanner system, force feedback from the shifter to the crane's drivetrain, as well as force feedback from the crane to the joystick system.

2 Meaning Analysis of Commands and Messages

The concept of the innovative intelligent interaction system includes a subsystem of recognition of speech commands in a natural language using patterns and antipatterns of commands, which is presented in Fig. 3. In the subsystem,

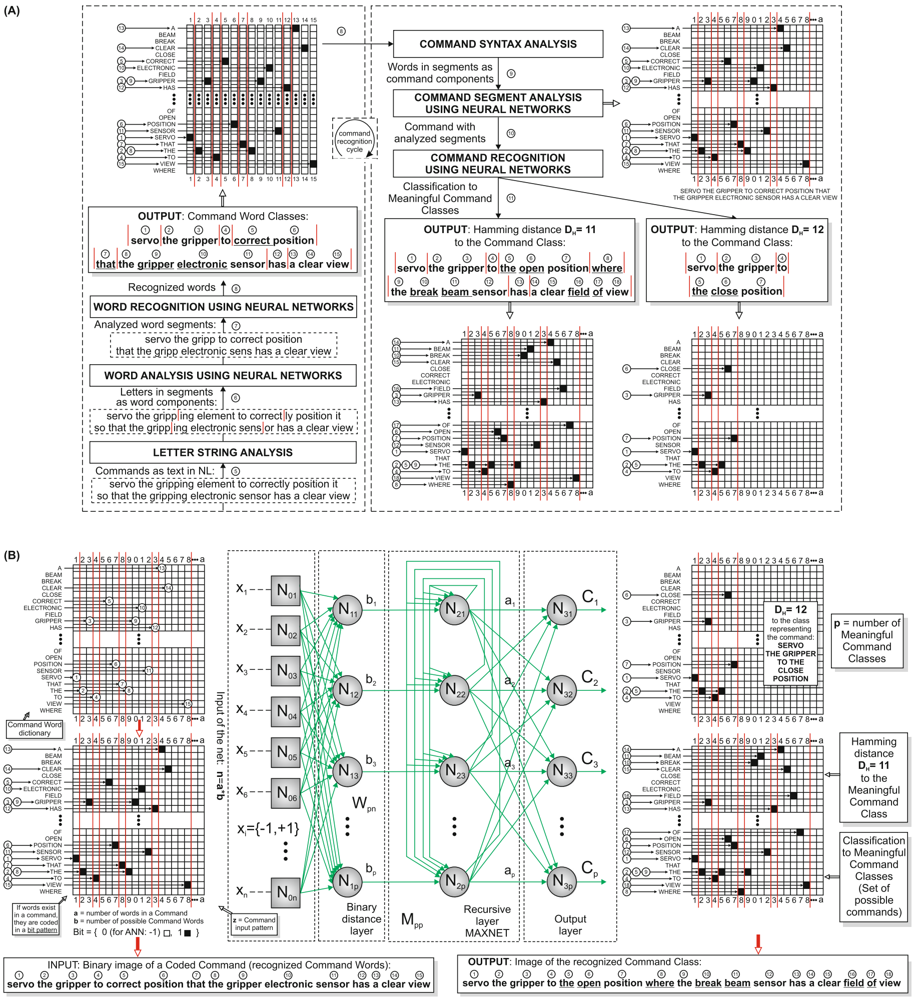

(A)
COMMAND SYNTAX ANALYSIS
Words in segments as command components
COMMAND SEGMENT ANALYSIS USING NEURAL NETWORKS
Command with analyzed segments
COMMAND RECOGNITION USING NEURAL NETWORKS
Classification to Meaningful Command Classes
command recognition cycle
OUTPUT: Command Word Classes:
servo the gripper to correct position
that the gripper electronic sensor has a clear view
Recognized words
WORD RECOGNITION USING NEURAL NETWORKS
Analyzed word segments:
servo the gripp to correct position
that the gripp electronic sens has a clear view
WORD ANALYSIS USING NEURAL NETWORKS
Letters in segments as word components:
servo the gripp|ing element to correct|ly position it
so that the gripp|ing electronic sens|or has a clear view
LETTER STRING ANALYSIS
Commands as text in NL:
servo the gripping element to correctly position it
so that the gripping electronic sensor has a clear view
OUTPUT: Hamming distance D_H= 11 to the Command Class:
servo the gripper to the open position where
the break beam sensor has a clear field of view
OUTPUT: Hamming distance D_H= 12 to the Command Class:
servo the gripper to
the close position
SERVO THE GRIPPER TO CORRECT POSITION THAT THE GRIPPER ELECTRONIC SENSOR HAS A CLEAR VIEW
A
BEAM
BREAK
CLEAR
CLOSE
CORRECT
ELECTRONIC
FIELD
GRIPPER
HAS
OF
OPEN
POSITION
SENSOR
SERVO
THAT
THE
TO
VIEW
WHERE
(B)
Command Word dictionary
If words exist in a command, they are coded in a bit pattern
a = number of words in a Command
b = number of possible Command Words
Bit = { 0 (for ANN: -1) □, 1 ■ }
z = Command input pattern
Input of the net: n=a*b
x_1 … x_n
x_i={-1,+1}
N_{01} … N_{0n}
N_{11} … N_{1p}
b_1 … b_p
W_{pn}
Binary distance layer
M_{pp}
N_{21} … N_{2p}
a_1 … a_p
Recursive layer MAXNET
N_{31} … N_{3p}
C_1 … C_p
Output layer
D_H= 12 to the class representing the command: SERVO THE GRIPPER TO THE CLOSE POSITION
p = number of Meaningful Command Classes
Hamming distance D_H= 11 to the Meaningful Command Class
Classification to Meaningful Command Classes (Set of possible commands)
INPUT: Binary image of a Coded Command (recognized Command Words):
servo the gripper to correct position that the gripper electronic sensor has a clear view
OUTPUT: Image of the recognized Command Class:
servo the gripper to the open position where the break beam sensor has a clear field of view

Fig. 4. (A) Block diagram of a meaning analysis cycle of an exemplary command, (B) Illustrative example of recognition of commands using binary neural networks.

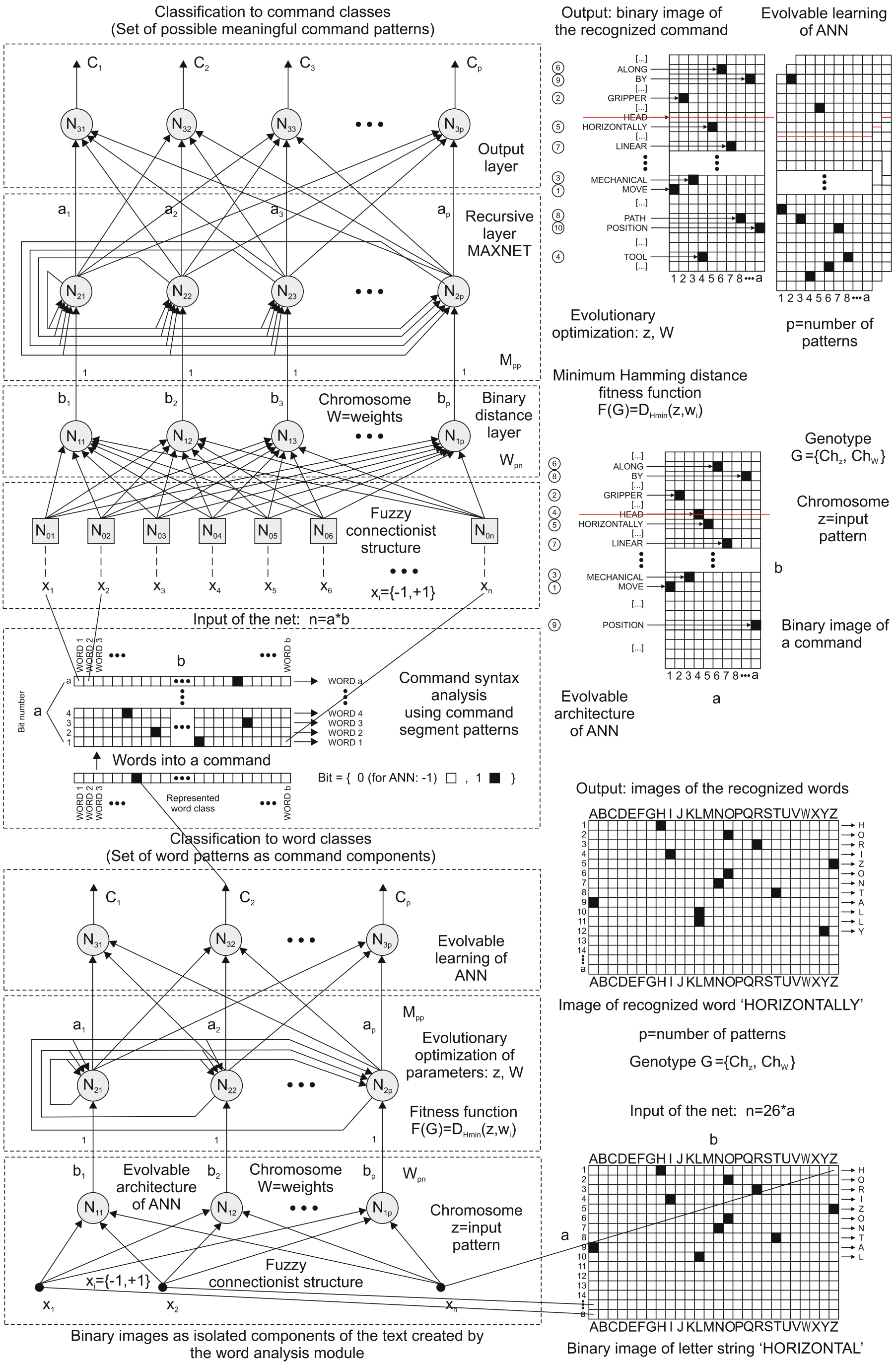
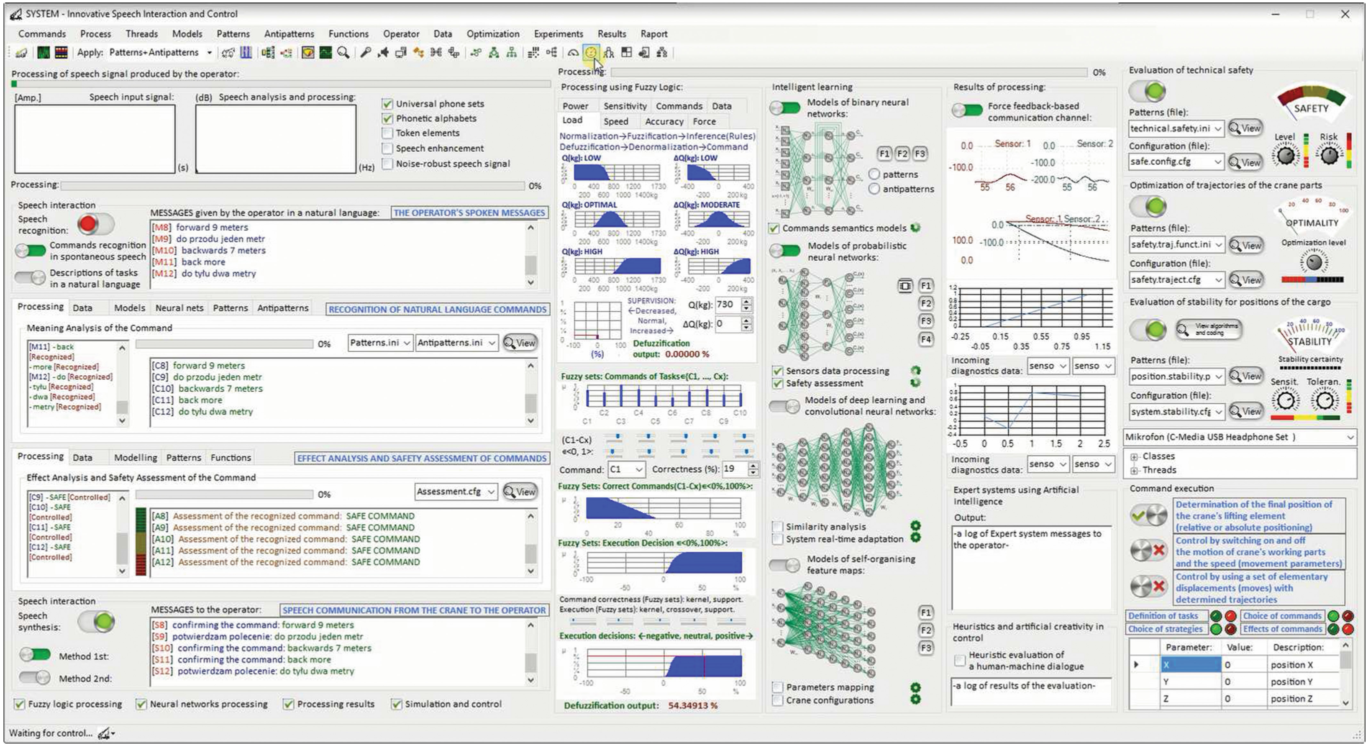

Classification to command classes
(Set of possible meaningful command patterns)
C1 C2 C3 Cp
N31 N32 N33 N3p
Output layer
a1 a2 a3 ap
Recursive layer MAXNET
N21 N22 N23 N2p
Mpp
b1 b2 b3 bp
Chromosome W=weights
Binary distance layer
N11 N12 N13 N1p
Wpn
N01 N02 N03 N04 N05 N06 N0n
Fuzzy connectionist structure
x1 x2 x3 x4 x5 x6 xn
xi={-1,+1}
Input of the net: n=a*b
WORD 1 WORD 2 WORD 3 WORD b
b
a
Bit number
WORD a
WORD 4
WORD 3
WORD 2
WORD 1
Command syntax analysis using command segment patterns
Words into a command
Bit = { 0 (for ANN: -1) □ , 1 ■ }
Represented word class
Output: binary image of the recognized command
Evolvable learning of ANN
ALONG
BY
GRIPPER
HEAD
HORIZONTALLY
LINEAR
MECHANICAL
MOVE
PATH
POSITION
TOOL
[...]
1 2 3 4 5 6 7 8 ••• a
Evolutionary optimization: z, W
p=number of patterns
Minimum Hamming distance fitness function
F(G)=DHmin(z,wi)
Genotype G={Chz, Chw}
Chromosome z=input pattern
b
Binary image of a command
Evolvable architecture of ANN
a
Classification to word classes
(Set of word patterns as command components)
C1 C2 Cp
N31 N32 N3p
Evolvable learning of ANN
Mpp
a1 a2 ap
Evolutionary optimization of parameters: z, W
N21 N22 N2p
Fitness function
F(G)=DHmin(z,wi)
b1 b2 bp
Evolvable architecture of ANN
Chromosome W=weights
Wpn
N11 N12 N1p
Chromosome z=input pattern
xi={-1,+1}
Fuzzy connectionist structure
x1 x2 xn
Binary images as isolated components of the text created by the word analysis module
Output: images of the recognized words
ABCDEFGHIJKLMNOPQRSTUVWXYZ
H O R I Z O N T A L L Y
Image of recognized word 'HORIZONTALLY'
p=number of patterns
Genotype G={Chz, Chw}
Input of the net: n=26*a
b
a
H O R I Z O N T A L
Binary image of letter string 'HORIZONTAL'

Fig. 5. Evolvable fuzzy neural networks for word and command recognition.


SYSTEM - Innovative Speech Interaction and Control
Commands Process Threads Models Patterns Antipatterns Functions Operator Data Optimization Experiments Results Raport
Apply: Patterns+Antipatterns
View
F1 F2 F3 F4

Processing of speech signal produced by the operator:

[Amp.] Speech input signal: (s)

(dB) Speech analysis and processing: (Hz)

☑ Universal phone sets
☑ Phonetic alphabets
□ Token elements
□ Speech enhancement
□ Noise-robust speech signal

Processing: 0%

Speech interaction
Speech recognition:
Commands recognition in spontaneous speech
Descriptions of tasks in a natural language

MESSAGES given by the operator in a natural language: THE OPERATOR'S SPOKEN MESSAGES

[M8] forward 9 meters
[M9] do przodu jeden metr
[M10] backwards 7 meters
[M11] back more
[M12] do tyłu dwa metry

Processing | Data | Models | Neural nets | Patterns | Antipatterns — RECOGNITION OF NATURAL LANGUAGE COMMANDS

Meaning Analysis of the Command — 0% — Patterns.ini — Antipatterns.ini

[M11] - back [Recognized]
- more [Recognized]
[M12] - do [Recognized]
- tyłu [Recognized]
- dwa [Recognized]
- metry [Recognized]

[C8] forward 9 meters
[C9] do przodu jeden metr
[C10] backwards 7 meters
[C11] back more
[C12] do tyłu dwa metry

Processing | Data | Modelling | Patterns | Functions — EFFECT ANALYSIS AND SAFETY ASSESSMENT OF COMMANDS

Effect Analysis and Safety Assessment of the Command — 0% — Assessment.cfg

[C9] - SAFE [Controlled]
[C10] - SAFE [Controlled]
[C11] - SAFE [Controlled]
[C12] - SAFE [Controlled]

[A8] Assessment of the recognized command: SAFE COMMAND
[A9] Assessment of the recognized command: SAFE COMMAND
[A10] Assessment of the recognized command: SAFE COMMAND
[A11] Assessment of the recognized command: SAFE COMMAND
[A12] Assessment of the recognized command: SAFE COMMAND

Speech interaction
Speech synthesis:
Method 1st:
Method 2nd:

MESSAGES to the operator: SPEECH COMMUNICATION FROM THE CRANE TO THE OPERATOR

[S8] confirming the command: forward 9 meters
[S9] potwierdzam polecenie: do przodu jeden metr
[S10] confirming the command: backwards 7 meters
[S11] confirming the command: back more
[S12] potwierdzam polecenie: do tyłu dwa metry

☑ Fuzzy logic processing ☑ Neural networks processing ☑ Processing results ☑ Simulation and control

Waiting for control...

Processing: 0%

Processing using Fuzzy Logic:
Power | Sensitivity | Commands | Data | Load | Speed | Accuracy | Force

Normalization→Fuzzification→Inference(Rules)
Defuzzification→Denormalization→Command

Q(kg): LOW — ΔQ(kg): LOW
Q(kg): OPTIMAL — ΔQ(kg): MODERATE
Q(kg): HIGH — ΔQ(kg): HIGH

SUPERVISION: ←Decreased, Normal, Increased→
Q(kg): 730
ΔQ(kg): 0
Defuzzification output: 0.00000 %

Fuzzy sets: Commands of Tasks∈{C1, ..., Cx}:

(C1-Cx) ∈<0, 1>:
Command: C1 Correctness (%): 19

Fuzzy Sets: Correct Commands(C1-Cx)∈<0%,100%>:

Fuzzy Sets: Execution Decision ∈<0%,100%>:

Command correctness (Fuzzy sets): kernel, support.
Execution (Fuzzy sets): kernel, crossover, support.

Execution decisions: ←negative, neutral, positive→

Defuzzification output: 54.34913 %

Intelligent learning
Models of binary neural networks:
○ patterns
○ antipatterns
☑ Commands semantics models
Models of probabilistic neural networks:
☑ Sensors data processing
☑ Safety assessment
Models of deep learning and convolutional neural networks:
□ Similarity analysis
□ System real-time adaptation
Models of self-organising feature maps:
□ Parameters mapping
□ Crane configurations

Results of processing:
Force feedback-based communication channel:
Sensor: 1 Sensor: 2
Incoming diagnostics data: senso senso
Incoming diagnostics data: senso senso

Expert systems using Artificial Intelligence
Output:
-a log of Expert system messages to the operator-

Heuristics and artificial creativity in control
□ Heuristic evaluation of a human-machine dialogue
-a log of results of the evaluation-

Evaluation of technical safety
Patterns (file): technical.safety.ini
Configuration (file): safe.config.cfg
SAFETY Level Risk

Optimization of trajectories of the crane parts
Patterns (file): safety.traj.funct.ini
Configuration (file): safety.traject.cfg
OPTIMALITY Optimization level

Evaluation of stability for positions of the cargo
View algorithms and coding
Patterns (file): position.stability.p
Configuration (file): system.stability.cfg
STABILITY Stability certainty Sensit. Toleran.

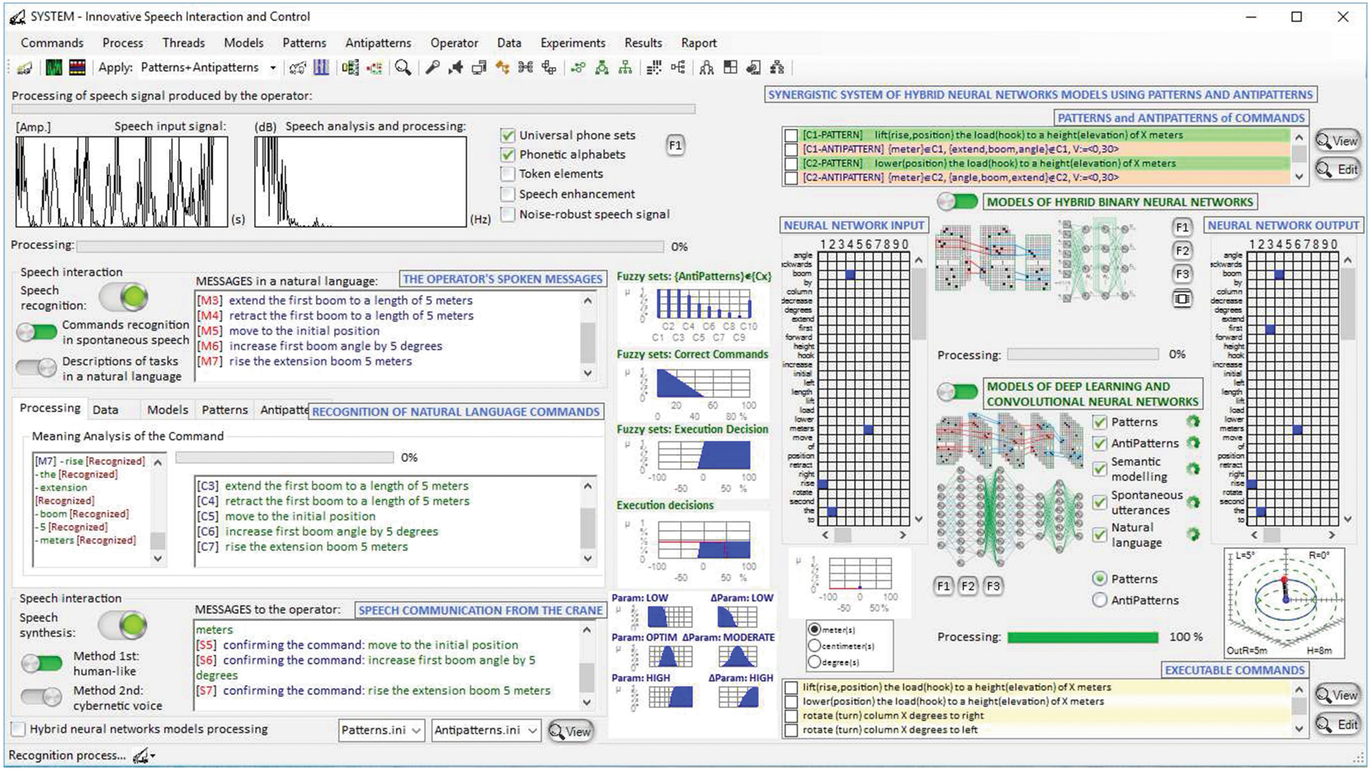
Mikrofon (C-Media USB Headphone Set)
Classes
Threads

Command execution
Determination of the final position of the crane's lifting element (relative or absolute positioning)
Control by switching on and off the motion of crane's working parts and the speed (movement parameters)
Control by using a set of elementary displacements (moves) with determined trajectories

Definition of tasks | Choice of commands
Choice of strategies | Effects of commands

Parameter:	Value:	Description:
X	0	position X
Y	0	position Y
Z	0	position Z

Fig. 6. The implementation of the innovative intelligent interaction system allowing different configurations and settings.


SYSTEM - Innovative Speech Interaction and Control
Commands
Process
Threads
Models
Patterns
Antipatterns
Operator
Data
Experiments
Results
Raport
Apply: Patterns+Antipatterns
Processing of speech signal produced by the operator:
[Amp.]
Speech input signal:
(s)
(dB)
Speech analysis and processing:
(Hz)
F1
Processing:
0%
SYNERGISTIC SYSTEM OF HYBRID NEURAL NETWORKS MODELS USING PATTERNS AND ANTIPATTERNS
PATTERNS and ANTIPATTERNS of COMMANDS
View
Edit
MODELS OF HYBRID BINARY NEURAL NETWORKS
NEURAL NETWORK INPUT
NEURAL NETWORK OUTPUT
F1
F2
F3
Speech interaction
Speech recognition:
Commands recognition in spontaneous speech
Descriptions of tasks in a natural language
MESSAGES in a natural language:
THE OPERATOR'S SPOKEN MESSAGES
Fuzzy sets: {AntiPatterns}∈{Cx}
C2 C4 C6 C8 C10
C1 C3 C5 C7 C9
Fuzzy sets: Correct Commands
Fuzzy sets: Execution Decision
Execution decisions
Processing
Data
Models
Patterns
Antipatte
RECOGNITION OF NATURAL LANGUAGE COMMANDS
Meaning Analysis of the Command
MODELS OF DEEP LEARNING AND CONVOLUTIONAL NEURAL NETWORKS
Speech interaction
Speech synthesis:
Method 1st: human-like
Method 2nd: cybernetic voice
MESSAGES to the operator:
SPEECH COMMUNICATION FROM THE CRANE
Param: LOW
ΔParam: LOW
Param: OPTIM
ΔParam: MODERATE
Param: HIGH
ΔParam: HIGH
L=5°
R=0°
OutR=5m
H=8m
EXECUTABLE COMMANDS
Processing:
100 %
Patterns.ini
Antipatterns.ini
View
Recognition process...

☑ Universal phone sets
☑ Phonetic alphabets
□ Token elements
□ Speech enhancement
□ Noise-robust speech signal

□ [C1-PATTERN] lift(rise,position) the load(hook) to a height(elevation) of X meters
□ [C1-ANTIPATTERN] {meter}∈C1, {extend,boom,angle}∉C1, V:=<0,30>
□ [C2-PATTERN] lower(position) the load(hook) to a height(elevation) of X meters
□ [C2-ANTIPATTERN] {meter}∈C2, {angle,boom,extend}∉C2, V:=<0,30>

[M3] extend the first boom to a length of 5 meters
[M4] retract the first boom to a length of 5 meters
[M5] move to the initial position
[M6] increase first boom angle by 5 degrees
[M7] rise the extension boom 5 meters

[M7] - rise [Recognized]
- the [Recognized]
- extension [Recognized]
- boom [Recognized]
- 5 [Recognized]
- meters [Recognized]

[C3] extend the first boom to a length of 5 meters
[C4] retract the first boom to a length of 5 meters
[C5] move to the initial position
[C6] increase first boom angle by 5 degrees
[C7] rise the extension boom 5 meters

meters
[S5] confirming the command: move to the initial position
[S6] confirming the command: increase first boom angle by 5 degrees
[S7] confirming the command: rise the extension boom 5 meters

☑ Patterns
☑ AntiPatterns
☑ Semantic modelling
☑ Spontaneous utterances
☑ Natural language

● Patterns
○ AntiPatterns

● meter(s)
○ centimeter(s)
○ degree(s)

□ lift(rise,position) the load(hook) to a height(elevation) of X meters
□ lower(position) the load(hook) to a height(elevation) of X meters
□ rotate (turn) column X degrees to right
□ rotate (turn) column X degrees to left

□ Hybrid neural networks models processing

Fig. 7. The implementation of the innovative intelligent interaction system using patterns and antipatterns.

the speech signal is converted to text and numerical values by the continuous speech recognition module [14,15]. After a successful utterance recognition, a text command in a natural language is further processed.

Individual words treated as isolated components of the text are subsequently processed with the modules for lexical analysis, tokenization and parsing [16–18]. After the text analysis, the letters grouped in segments are processed by the word analysis module. In the next stage, the analyzed word segments are inputs of the neural network for recognizing words. The network uses a training file containing also words and is trained to recognize words as command components, with words represented by output neurons.

In the meaning analysis process of text commands (Fig. 4A) in a natural language, the meaning analysis of words as command or message components is performed [19]. The recognized words are transferred to the command syntax analysis module which uses command segment patterns.

It analyses commands and identifies them as segments with regards to meaning, and also codes commands as vectors. They are sent to the command segment analysis module using encoded command segment patterns. The commands become inputs of the command recognition module using neural networks. The module uses a 3-layer Hamming network to classify the command and find its meaning (Fig. 4B). The neural network of this module uses a training file with possible meaningful commands.

The proposed method for meaning analysis of words, commands and messages uses binary neural networks (Fig. 4A, B) for natural language understanding. The motivation behind using this type of neural networks for meaning analysis [19] is that they offer an advantage of simple binarization of words, commands and sentences, as well as very fast training and run-time response. The cycle of exemplary command meaning analysis is presented in Fig. 4A. The proposed concept of processing of words and messages enables a variety of analyses of the spoken commands in a natural language. The developed methods for word and command recognition feature evolvable fuzzy neural networks (Fig. 5).

The experimental implementation of the innovative intelligent interaction system between a loader crane and its human operator using spoken natural languages is presented in Figs. 6 and 7. The complex system also features processing data using patterns and antipatterns of commands.

3 Conclusions and Perspectives

The designed interaction system is equipped with the most modern artificial intelligence-based technologies: voice communication, vision systems, augmented reality and interactive manipulators with force feedback. Modern control and supervision systems allow to efficiently and securely transfer, and precisely place materials, products and fragile cargo. The proposed design of the innovative AR speech interface for controlling lifting devices has been based on hybrid neural network architectures. The design can be considered as an attempt to create a new standard of the intelligent system for execution, control, supervision and optimization of effective and flexible cargo manipulation processes using communication by speech and natural language.

Acknowledgements. This project is financed by the National Centre for Research and Development, Poland (NCBiR), under the Applied Research Programme - Grant agreement No. PBS3/A6/28/2015.

References

1. Kumar, A., Metze, F., Kam, M.: Enabling the rapid development and adoption of speech-user interfaces. Computer **47**(1), 40–47 (2014). IEEE
2. Ortiz, C.L.: The road to natural conversational speech interfaces. IEEE Internet Comput. **18**(2), 74–78 (2014). IEEE
3. Majewski, M., Kacalak, W.: Conceptual design of innovative speech interfaces with augmented reality and interactive systems for controlling loader cranes. In: Silhavy, R., Senkerik, R., Oplatkova, Z.K., Silhavy, P., Prokopova, Z. (eds.) Artificial Intelligence Perspectives in Intelligent Systems. AISC, vol. 464, pp. 237–247. Springer, Switzerland (2016). doi:10.1007/978-3-319-33625-1_22
4. Majewski, M., Kacalak, W.: Intelligent speech interaction of devices and human operators. In: Silhavy, R., Senkerik, R., Oplatkova, Z.K., Silhavy, P., Prokopova, Z. (eds.) Software Engineering Perspectives and Application in Intelligent Systems. AISC, vol. 465, pp. 471–482. Springer, Switzerland (2016). doi:10.1007/978-3-319-33622-0_42
5. Kacalak, W., Majewski, M.: New intelligent interactive automated systems for design of machine elements and assemblies. In: Huang, T., Zeng, Z., Li, C., Leung, C.S. (eds.) ICONIP 2012. LNCS, vol. 7666, pp. 115–122. Springer, Heidelberg (2012). doi:10.1007/978-3-642-34478-7_15
6. Kacalak, W., Majewski, M., Budniak, Z.: Interactive systems for designing machine elements and assemblies. Manag. Prod. Eng. Rev. **6**(3), 21–34 (2015). De Gruyter Open
7. Kacalak, W., Majewski, M., Budniak, Z.: Intelligent automated design of machine components using antipatterns. In: Jackowski, K., Burduk, R., Walkowiak, K., Woźniak, M., Yin, H. (eds.) IDEAL 2015. LNCS, vol. 9375, pp. 248–255. Springer, Heidelberg (2015). doi:10.1007/978-3-319-24834-9_30
8. Pajor, M., Grudziski, M.: Intelligent machine tool - vision based 3D scanning system for positioning of the Workpiece. Solid State Phenom. **220–221**, 497–503 (2015)
9. Pajor, M., Miadlicki, K.: Real-time gesture control of a CNC machine tool with the use microsoft kinect sensor. J. Sci. Eng. Res. **6**(9), 538–543 (2015)
10. Pajor, M., Miadlicki, K.: Overview of user interfaces used in load lifting devices. J. Sci. Eng. Res. **6**(9), 1215–1220 (2015)
11. Kacalak, W., Budniak, Z., Majewski, M.: Crane stability for various load conditions and trajectories of load translocation. Mechanik **2016**(12), 1820–1823 (2016). doi:10.17814/mechanik.2016.12.571
12. Kacalak, W., Budniak, Z., Majewski, M.: Simulation model of a mobile crane with ensuring its stability. Model. Eng. **29**(60), 35–43 (2016). PTMTS
13. Taylor, P.: Text-to-Speech Synthesis. Cambridge University Press, Cambridge (2008)

14. Oppenheim, A.V., Schafer, R.W.: Discrete - Time Signal Processing. Prentice-Hall, Upper Saddle River (2010)
15. Buck, J.R., Daniel, M.M., Singer, A.C.: Computer Explorations in Signals and Systems Using Matlab. Prentice Hall, Upper Saddle River (2002)
16. Stuart, K.D., Majewski, M., Trelis, A.B.: Selected problems of intelligent corpus analysis through probabilistic neural networks. In: Zhang, L., Lu, B.-L., Kwok, J. (eds.) ISNN 2010. LNCS, vol. 6064, pp. 268–275. Springer, Heidelberg (2010). doi:10.1007/978-3-642-13318-3_34
17. Stuart, K.D., Majewski, M., Trelis, A.B.: Intelligent semantic-based system for corpus analysis through hybrid probabilistic neural networks. In: Liu, D., Zhang, H., Polycarpou, M., Alippi, C., He, H. (eds.) ISNN 2011, Part I. LNCS, vol. 6675, pp. 83–92. Springer, Heidelberg (2011). doi:10.1007/978-3-642-21105-8_11
18. Stuart, K.D., Majewski, M.: Intelligent opinion mining and sentiment analysis using artificial neural networks. In: Arik, S., Huang, T., Lai, W.K., Liu, Q. (eds.) ICONIP 2015, Part IV. LNCS, vol. 9492, pp. 103–110. Springer, Switzerland (2015). doi:10.1007/978-3-319-26561-2_13
19. Majewski, M., Zurada, J.M.: Sentence recognition using artificial neural networks. Knowl Based Syst. **21**(7), 629–635 (2008). Elsevier

Smart Control of Lifting Devices Using Patterns and Antipatterns

Maciej Majewski(✉) and Wojciech Kacalak

Faculty of Mechanical Engineering, Koszalin University of Technology,
Raclawicka 15-17, 75-620 Koszalin, Poland
{maciej.majewski,wojciech.kacalak}@tu.koszalin.pl

Abstract. The paper presents a concept of smart lifting devices with speech-based interaction using patterns and antipatterns. It proposes reasoning models for inference mechanism of implicit relations between commands, tasks and processes in the interactive crane control. The developed models can be used in automation of command recognition, safety analysis and assessment in human-machine interaction. In the concept, commands, tasks and processes are processed using probabilistic neural networks and neural classifiers. The commands provide information about actions and objects. The data about tasks are based on operations and targets. The processes depend on parameters and working conditions. As the result, the extracted structured data with associated information allows for development of patterns and antipatterns for the crane control and safety analysis by uncovering implicit relations between commands, tasks and processes.

Keywords: Smart control · Interactive system · Speech communication · Reasoning models · Antipatterns · Neural networks · Artificial intelligence · Innovative Interfaces · Lifting devices

1 Smart Lifting Devices

Lifting devices are tremendously utilized in numerous heavy load transportation industries, and therefore, the smart control of crane systems is becoming an interesting and important research field. Innovative control systems designed for processes of precise positioning of objects and heavy cargo can be equipped with intelligent interaction systems between lifting devices and their human operators.

Third-generation smart control systems combine artificial intelligence and cognitive functions so that they can provide an interface between the operator's augmented reality vision and the physical lifting device and its environment. The systems for smart control of lifting devices typically consist of diverse components: tracking cameras, augmented reality goggles, laser trackers, speech and natural language processing methods, geometry and topography scanners, photogrammetric cameras, movement scanners, interactive manipulators, measurement tools, sensors for signal acquisition, actuators for performing or triggering

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_48

actions. The smart control systems address natural interfaces, speech interaction [1], interactive communication, vision systems, augmented reality, interactive manipulators, feedback channels, force feedback, and distributed control.

Smart control of lifting devices incorporate functions of sensing of device working conditions and environment, actuation and control of mechanical and mechatronic systems, in order to analyze and model a task or situation, and make decisions based on the acquired data in a predictive or adaptive manner, thereby performing smart control actions for the lifting device and processes. In general the smartness of the control system can be attributed to intelligent control operation based on augmented reality vision, natural interfaces, sensorial systems, increased safety, and cooperation capabilities. Advanced smart control systems address safety, efficiency and ergonomic challenges like automation, optimization, supervision, flexibility, adaptability, robustness, and the ability to reuse knowledge. They are for that reason increasingly used in a large number of different tasks. Key sectors in this context are construction, transportation, logistics and manufacturing.

2 Smart Control Systems with Innovative Interfaces

The proposed conceptual design of smart control systems assumes that they feature natural-language speech interfaces, intelligent visual-aid systems based on augmented reality, as well as interactive manipulation systems providing force feedback. A sketch of the technical and scientific problem space facing a scalable and universal realization of the smart control system is shown in Fig. 1.

The aim of the presented research is to design an innovative human-machine speech-based interface using natural languages for smart lifting devices. The proposed interface [2–5] equipped with patterns and antipatterns and artificial intelligence methods allows for safety and performance improvement. This especially applies when the expert assistance is needed regarding distant effects and complex decisions, heuristic circumstances of decisions, and sudden changes of conditions. This knowledge can be used for distributed smart control.

The innovative design of the smart control system of lifting devices involves the use of an intelligent and natural interface applying patterns and antipatterns for automatic processing of commands, tasks and processes in order to extract hidden information through uncovering implicit relations for various dependencies between tasks, conditions, processes and parameters. Another approach of exploration of implicit relations concerns mining in meaning correlations between commands, tasks and strategies, using patterns and antipatterns.

The design and implementation of the control system with innovative interfaces for smart lifting devices is based on simulations and experiments with developed kinematic models of loader cranes. They allowed for the development of intelligent methods for processing data in operating cycles of the control process. An analysis of the configuration system of the crane's loading system was carried out in order to ensure collision-free motion of parts, as well as kinematics models for different tasks were developed. Moreover a set of tasks for the crane

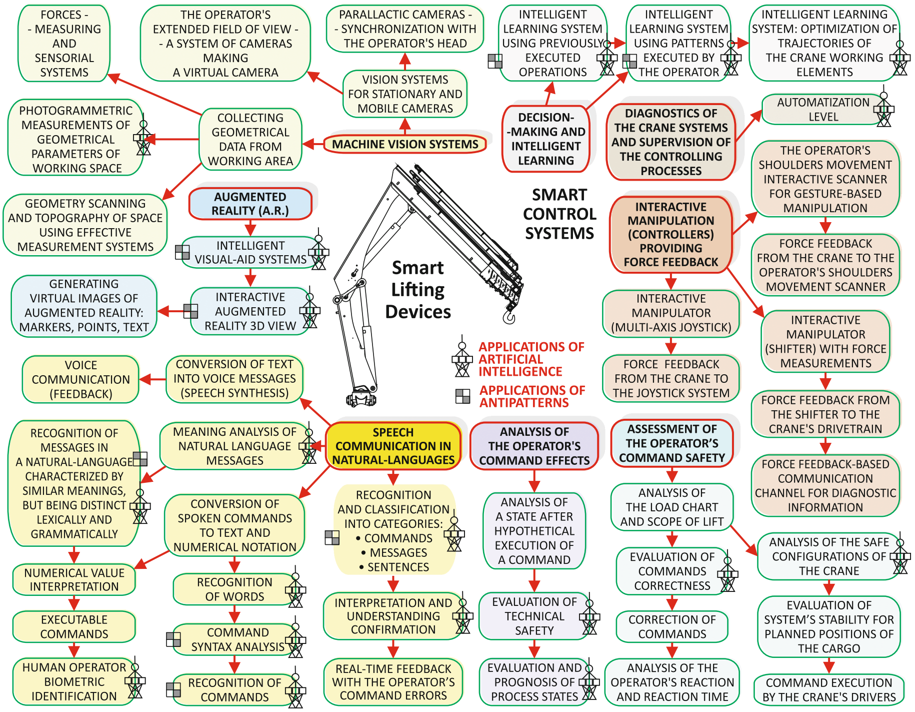

FORCES - - MEASURING AND SENSORIAL SYSTEMS
THE OPERATOR'S EXTENDED FIELD OF VIEW - - A SYSTEM OF CAMERAS MAKING A VIRTUAL CAMERA
PARALLACTIC CAMERAS - - SYNCHRONIZATION WITH THE OPERATOR'S HEAD
INTELLIGENT LEARNING SYSTEM USING PREVIOUSLY EXECUTED OPERATIONS
INTELLIGENT LEARNING SYSTEM USING PATTERNS EXECUTED BY THE OPERATOR
INTELLIGENT LEARNING SYSTEM: OPTIMIZATION OF TRAJECTORIES OF THE CRANE WORKING ELEMENTS
VISION SYSTEMS FOR STATIONARY AND MOBILE CAMERAS
PHOTOGRAMMETRIC MEASUREMENTS OF GEOMETRICAL PARAMETERS OF WORKING SPACE
COLLECTING GEOMETRICAL DATA FROM WORKING AREA
MACHINE VISION SYSTEMS
DECISION- -MAKING AND INTELLIGENT LEARNING
DIAGNOSTICS OF THE CRANE SYSTEMS AND SUPERVISION OF THE CONTROLLING PROCESSES
AUTOMATIZATION LEVEL
THE OPERATOR'S SHOULDERS MOVEMENT INTERACTIVE SCANNER FOR GESTURE-BASED MANIPULATION
GEOMETRY SCANNING AND TOPOGRAPHY OF SPACE USING EFFECTIVE MEASUREMENT SYSTEMS
AUGMENTED REALITY (A.R.)
INTELLIGENT VISUAL-AID SYSTEMS
SMART CONTROL SYSTEMS
INTERACTIVE MANIPULATION (CONTROLLERS) PROVIDING FORCE FEEDBACK
FORCE FEEDBACK FROM THE CRANE TO THE OPERATOR'S SHOULDERS MOVEMENT SCANNER
Smart Lifting Devices
GENERATING VIRTUAL IMAGES OF AUGMENTED REALITY: MARKERS, POINTS, TEXT
INTERACTIVE AUGMENTED REALITY 3D VIEW
INTERACTIVE MANIPULATOR (MULTI-AXIS JOYSTICK)
INTERACTIVE MANIPULATOR (SHIFTER) WITH FORCE MEASUREMENTS
APPLICATIONS OF ARTIFICIAL INTELLIGENCE
APPLICATIONS OF ANTIPATTERNS
VOICE COMMUNICATION (FEEDBACK)
CONVERSION OF TEXT INTO VOICE MESSAGES (SPEECH SYNTHESIS)
FORCE FEEDBACK FROM THE CRANE TO THE JOYSTICK SYSTEM
FORCE FEEDBACK FROM THE SHIFTER TO THE CRANE'S DRIVETRAIN
RECOGNITION OF MESSAGES IN A NATURAL-LANGUAGE CHARACTERIZED BY SIMILAR MEANINGS, BUT BEING DISTINCT LEXICALLY AND GRAMMATICALLY
MEANING ANALYSIS OF NATURAL LANGUAGE MESSAGES
SPEECH COMMUNICATION IN NATURAL-LANGUAGES
ANALYSIS OF THE OPERATOR'S COMMAND EFFECTS
ASSESSMENT OF THE OPERATOR'S COMMAND SAFETY
FORCE FEEDBACK-BASED COMMUNICATION CHANNEL FOR DIAGNOSTIC INFORMATION
RECOGNITION AND CLASSIFICATION INTO CATEGORIES:
• COMMANDS
• MESSAGES
• SENTENCES
ANALYSIS OF A STATE AFTER HYPOTHETICAL EXECUTION OF A COMMAND
ANALYSIS OF THE LOAD CHART AND SCOPE OF LIFT
CONVERSION OF SPOKEN COMMANDS TO TEXT AND NUMERICAL NOTATION
ANALYSIS OF THE SAFE CONFIGURATIONS OF THE CRANE
EVALUATION OF COMMANDS CORRECTNESS
NUMERICAL VALUE INTERPRETATION
RECOGNITION OF WORDS
INTERPRETATION AND UNDERSTANDING CONFIRMATION
EVALUATION OF TECHNICAL SAFETY
EVALUATION OF SYSTEM'S STABILITY FOR PLANNED POSITIONS OF THE CARGO
EXECUTABLE COMMANDS
COMMAND SYNTAX ANALYSIS
CORRECTION OF COMMANDS
HUMAN OPERATOR BIOMETRIC IDENTIFICATION
RECOGNITION OF COMMANDS
REAL-TIME FEEDBACK WITH THE OPERATOR'S COMMAND ERRORS
EVALUATION AND PROGNOSIS OF PROCESS STATES
ANALYSIS OF THE OPERATOR'S REACTION AND REACTION TIME
COMMAND EXECUTION BY THE CRANE'S DRIVERS

Fig. 1. Map of the technical and scientific problem space for the smart control systems of lifting devices with application of artificial intelligence and antipatterns.

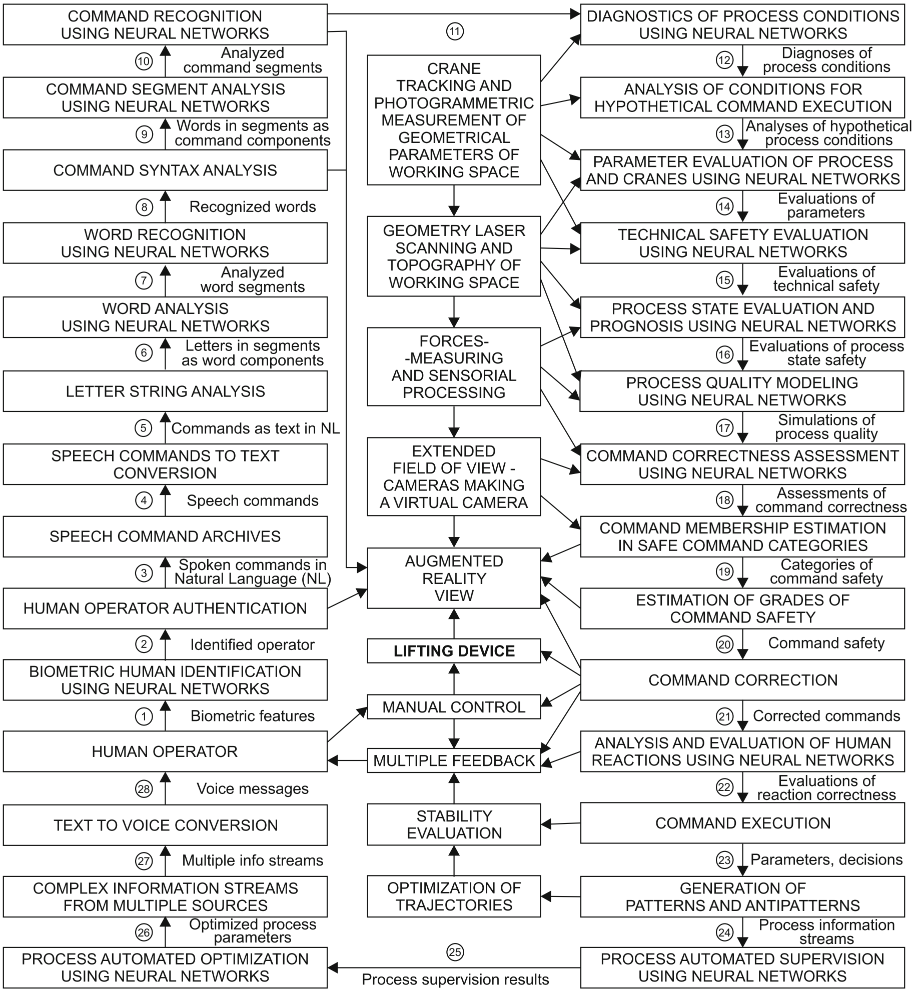
were devised, motion components for movable elements were determined, as well as ranges of motion and allowed trajectories depending on characteristics of executed tasks. In addition the patterns representing correct execution of tasks were devised, in the form of motion sequences of crane's working parts.

The analyzes were used to enable rigorous development of algorithms and software for modeling the manner of execution of selected operator's actions of controlling the motion of crane's working parts. The research also allowed for rigorous development of algorithms and software for modeling the motion of crane's working elements, which take into account components of distance from the target point.

3 Intelligent and Natural Interfaces

Smart lifting devices feature control systems with innovative interfaces using artificial intelligence methods and techniques for processing speech and natural language, as well as supporting decision-making processes. The concept of the interaction systems between lifting devices and their human operators assumes that they are equipped with the following subsystems: augmented reality vision, voice communication, natural language processing, command effect analysis, command safety assessment, command execution, supervision and diagnostics, decision-making and learning, interactive manipulation with force feedback.

The smart control system with an intelligent and natural interface is presented in abbreviated form in Fig. 2. The numbers in the cycle represent the successive phases of information processing. The system is equipped with specialized modules using the following real-time data sources: crane tracking and photogrammetric measurement of geometrical parameters of working space, laser scanning of geometry and topography of working space, forces-measuring and sensorial processing. The obtained data is sent to the subsystem of effect analysis



COMMAND RECOGNITION USING NEURAL NETWORKS
10 Analyzed command segments
COMMAND SEGMENT ANALYSIS USING NEURAL NETWORKS
9 Words in segments as command components
COMMAND SYNTAX ANALYSIS
8 Recognized words
WORD RECOGNITION USING NEURAL NETWORKS
7 Analyzed word segments
WORD ANALYSIS USING NEURAL NETWORKS
6 Letters in segments as word components
LETTER STRING ANALYSIS
5 Commands as text in NL
SPEECH COMMANDS TO TEXT CONVERSION
4 Speech commands
SPEECH COMMAND ARCHIVES
3 Spoken commands in Natural Language (NL)
HUMAN OPERATOR AUTHENTICATION
2 Identified operator
BIOMETRIC HUMAN IDENTIFICATION USING NEURAL NETWORKS
1 Biometric features
HUMAN OPERATOR
28 Voice messages
TEXT TO VOICE CONVERSION
27 Multiple info streams
COMPLEX INFORMATION STREAMS FROM MULTIPLE SOURCES
26 Optimized process parameters
PROCESS AUTOMATED OPTIMIZATION USING NEURAL NETWORKS
11
CRANE TRACKING AND PHOTOGRAMMETRIC MEASUREMENT OF GEOMETRICAL PARAMETERS OF WORKING SPACE
GEOMETRY LASER SCANNING AND TOPOGRAPHY OF WORKING SPACE
FORCES- -MEASURING AND SENSORIAL PROCESSING
EXTENDED FIELD OF VIEW - CAMERAS MAKING A VIRTUAL CAMERA
AUGMENTED REALITY VIEW
LIFTING DEVICE
MANUAL CONTROL
MULTIPLE FEEDBACK
STABILITY EVALUATION
OPTIMIZATION OF TRAJECTORIES
25
Process supervision results
DIAGNOSTICS OF PROCESS CONDITIONS USING NEURAL NETWORKS
12 Diagnoses of process conditions
ANALYSIS OF CONDITIONS FOR HYPOTHETICAL COMMAND EXECUTION
13 Analyses of hypothetical process conditions
PARAMETER EVALUATION OF PROCESS AND CRANES USING NEURAL NETWORKS
14 Evaluations of parameters
TECHNICAL SAFETY EVALUATION USING NEURAL NETWORKS
15 Evaluations of technical safety
PROCESS STATE EVALUATION AND PROGNOSIS USING NEURAL NETWORKS
16 Evaluations of process state safety
PROCESS QUALITY MODELING USING NEURAL NETWORKS
17 Simulations of process quality
COMMAND CORRECTNESS ASSESSMENT USING NEURAL NETWORKS
18 Assessments of command correctness
COMMAND MEMBERSHIP ESTIMATION IN SAFE COMMAND CATEGORIES
19 Categories of command safety
ESTIMATION OF GRADES OF COMMAND SAFETY
20 Command safety
COMMAND CORRECTION
21 Corrected commands
ANALYSIS AND EVALUATION OF HUMAN REACTIONS USING NEURAL NETWORKS
22 Evaluations of reaction correctness
COMMAND EXECUTION
23 Parameters, decisions
GENERATION OF PATTERNS AND ANTIPATTERNS
24 Process information streams
PROCESS AUTOMATED SUPERVISION USING NEURAL NETWORKS

Fig. 2. Smart control systems with intelligent and natural interfaces.

and safety assessment. The operator's augmented reality view is based on the extended field of view from cameras making a virtual camera. The lifting device is controlled using spoken natural language commands and manual control with multiple feedback. Developed new methods for stability evaluation and optimization of trajectories are also included in the smart control.

4 Models of Patterns and Antipatterns of Commands, Tasks and Processes

In the proposed concept, smart control systems feature developed methods of intelligent mining of structured data and information for generation of patterns and antipatterns in the interactive control process. The concept proposes reasoning models for inference mechanism of implicit relations between commands, tasks and processes. The proposed methodology is based on probabilistic neural networks [6] (Fig. 3A) and hybrid neural classifiers (Fig. 3B).

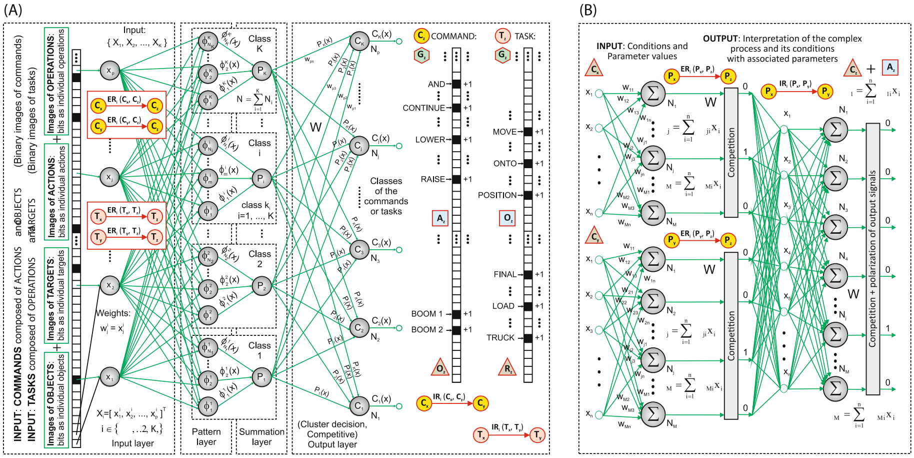
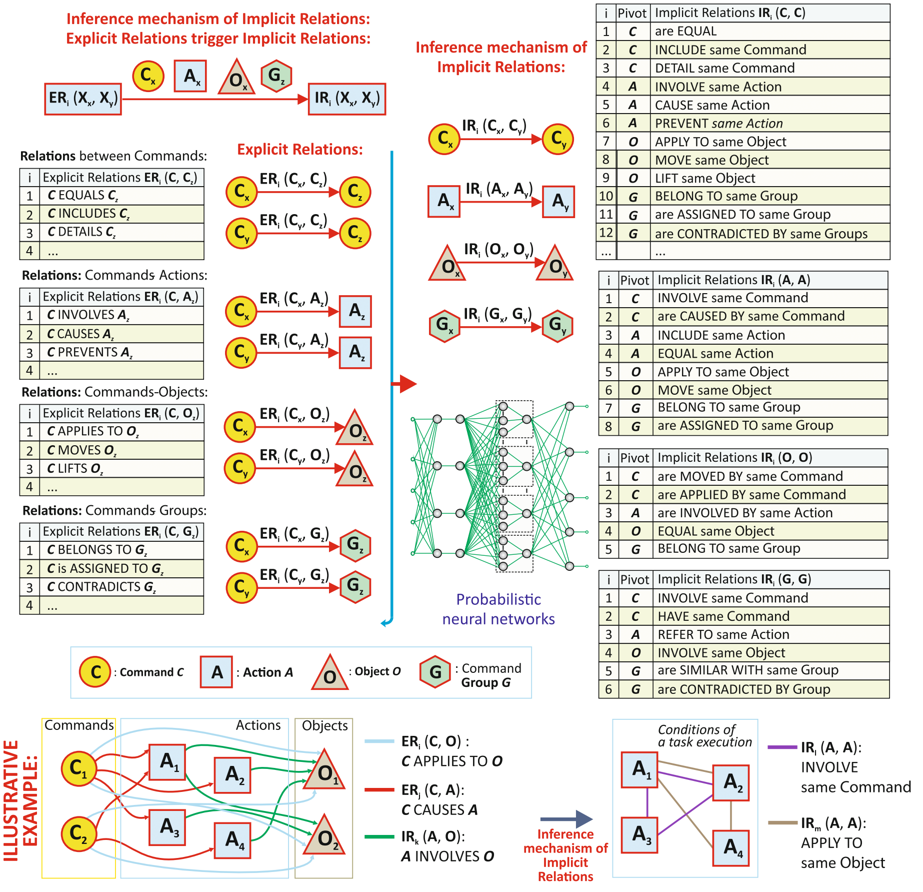

(A)
INPUT: COMMANDS composed of ACTIONS and OBJECTS (Binary images of commands)
INPUT: TASKS composed of OPERATIONS and TARGETS (Binary images of tasks)
Images of OPERATIONS: bits as individual operations
Images of ACTIONS: bits as individual actions
Images of TARGETS: bits as individual targets
Images of OBJECTS: bits as individual objects
Input: { X1, X2, ..., XK }
ERi (Cx, Cz)
ERi (Cy, Cz)
ERi (Tx, Tz)
ERi (Ty, Tz)
Weights: wij = xij
X=[x1, x2, ..., xp]T
i ∈ { ,..2, K,}
Input layer
Pattern layer
Summation layer
Class K
Class i
Class 2
Class 1
N = ∑ Ni
class ki i=1, ..., K
W
Classes of the commands or tasks
(Cluster decision, Competitive) Output layer
COMMAND:
TASK:
AND
CONTINUE
LOWER
RAISE
BOOM 1
BOOM 2
MOVE
ONTO
POSITION
FINAL
LOAD
TRUCK
+1
IRi (Cx, Cy)
IRi (Tx, Ty)
(B)
INPUT: Conditions and Parameter values
OUTPUT: Interpretation of the complex process and its conditions with associated parameters
ERi (Px, Pz)
ERi (Py, Pz)
IRi (Px, Py)
Competition
Competition + polarization of output signals
W

Fig. 3. Inference mechanism of implicit relations between commands, tasks and processes based on probabilistic neural networks (A) and hybrid neural classifiers (B).

The intelligent data mining methodology allows to determine associated commands with their connected actions and objects. It also provides associated tasks and their linked operations and targets. The networks discover relationships between actions and linked objects, and on this basis trigger execution of complete commands which cause the same actions and effects or control of machine assemblies e.g. boom systems. Mobile cranes feature boom systems which are composed of non-extendable booms, extendable telescoping booms, as well as sub-booms. An example can be the following command 'continue to raise the 2nd boom (telescoping boom) and lower the 1st boom' which causes the boom system to move the object (cargo) away from the crane. Another example could be the following command 'lower the 2nd boom and rise the 1st boom' which causes the object to be moved toward the crane. The methodology based on hybrid neural classifier allows for classification of process conditions and their relevant parameter values to condition groups for selected crane configurations. Inputs of the network comprise grouped parameters with values for each consecutive classification stage. The network's output produces interpretation of the complex process with its associated parameters for a specified conditions.

The developed methodology of modeling patterns and antipatterns for reasoning can be used in automation of command recognition, safety analysis and assessment in human-machine interaction. The methods allow for extraction of hidden information about the control process of operating lifting devices (cranes) through uncovering implicit relations for commands and their associated actions on objects (Fig. 4), tasks and their component operations on targets (Fig. 5A), as well as processes and their parameters with conditions (Fig. 5B).


Inference mechanism of Implicit Relations:
Explicit Relations trigger Implicit Relations:
C_x A_x O_x G_z
ER_i (X_x, X_y)
IR_i (X_x, X_y)
Explicit Relations:
ER_i (C_x, C_z)
ER_i (C_y, C_z)
ER_i (C_x, A_z)
ER_i (C_y, A_z)
ER_i (C_x, O_z)
ER_i (C_y, O_z)
ER_i (C_x, G_z)
ER_i (C_y, G_z)
Inference mechanism of Implicit Relations:
IR_i (C_x, C_y)
IR_i (A_x, A_y)
IR_i (O_x, O_y)
IR_i (G_x, G_y)
Probabilistic neural networks
C : Command *C*
A : Action *A*
O : Object *O*
G : Command Group *G*
ILLUSTRATIVE EXAMPLE:
Commands
Actions
Objects
C_1 C_2 A_1 A_2 A_3 A_4 O_1 O_2
ER_i (C, O) : *C* APPLIES TO *O*
ER_j (C, A): *C* CAUSES *A*
IR_k (A, O): *A* INVOLVES *O*
Inference mechanism of Implicit Relations
Conditions of a task execution
IR_l (A, A): INVOLVE same Command
IR_m (A, A): APPLY TO same Object

Relations between Commands:

i	Explicit Relations ER_i (C, C_z)
1	*C* EQUALS C_z
2	*C* INCLUDES C_z
3	*C* DETAILS C_z
4	...

Relations: Commands Actions:

i	Explicit Relations ER_i (C, A_z)
1	*C* INVOLVES A_z
2	*C* CAUSES A_z
3	*C* PREVENTS A_z
4	...

Relations: Commands-Objects:

i	Explicit Relations ER_i (C, O_z)
1	*C* APPLIES TO O_z
2	*C* MOVES O_z
3	*C* LIFTS O_z
4	...

Relations: Commands Groups:

i	Explicit Relations ER_i (C, G_z)
1	*C* BELONGS TO G_z
2	*C* is ASSIGNED TO G_z
3	*C* CONTRADICTS G_z
4	...

i	Pivot	Implicit Relations IR_i (C, C)
1	*C*	are EQUAL
2	*C*	INCLUDE same Command
3	*C*	DETAIL same Command
4	*A*	INVOLVE same Action
5	*A*	CAUSE same Action
6	*A*	PREVENT *same Action*
7	*O*	APPLY TO same Object
8	*O*	MOVE same Object
9	*O*	LIFT same Object
10	*G*	BELONG TO same Group
11	*G*	are ASSIGNED TO same Group
12	*G*	are CONTRADICTED BY same Groups
...		...

i	Pivot	Implicit Relations IR_i (A, A)
1	*C*	INVOLVE same Command
2	*C*	are CAUSED BY same Command
3	*A*	INCLUDE same Action
4	*A*	EQUAL same Action
5	*O*	APPLY TO same Object
6	*O*	MOVE same Object
7	*G*	BELONG TO same Group
8	*G*	are ASSIGNED TO same Group

i	Pivot	Implicit Relations IR_i (O, O)
1	*C*	are MOVED BY same Command
2	*C*	are APPLIED BY same Command
3	*A*	are INVOLVED BY same Action
4	*O*	EQUAL same Object
5	*G*	BELONG TO same Group

i	Pivot	Implicit Relations IR_i (G, G)
1	*C*	INVOLVE same Command
2	*C*	HAVE same Command
3	*A*	REFER TO same Action
4	*O*	INVOLVE same Object
5	*G*	are SIMILAR WITH same Group
6	*G*	are CONTRADICTED BY Group

Fig. 4. Modeling patterns and antipatterns for reasoning based on neural networks.

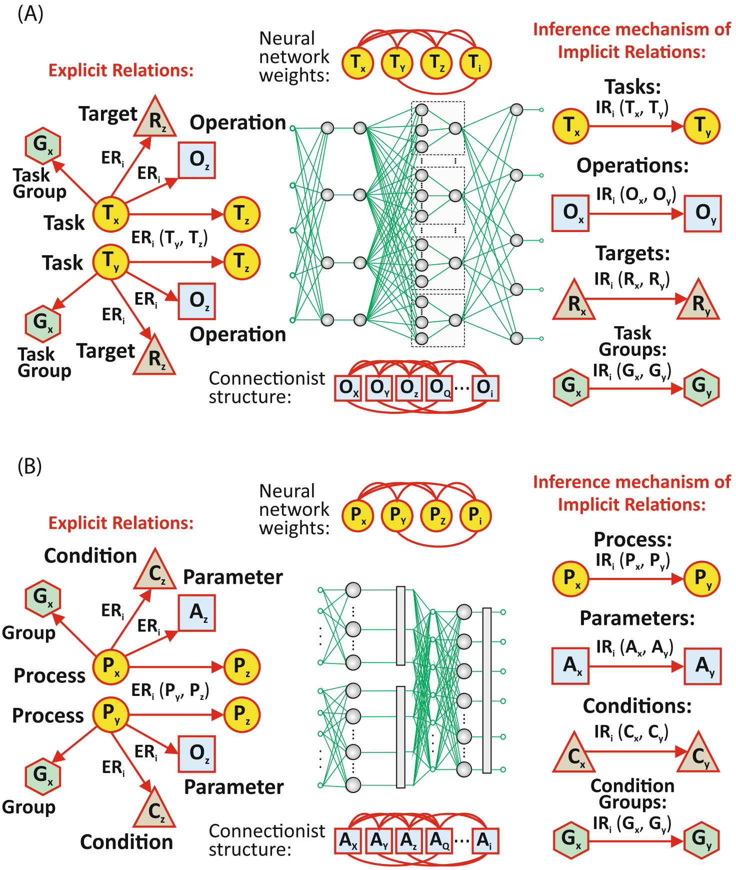


Fig. 5. Reasoning models for information mining based on neural networks to extract hidden information about tasks (A) and processes (B) through uncovering implicit relations for tasks and their component operations on targets, processes and their associated parameters with conditions.

In the concept, issued commands, given tasks (strategies) and executed processes are processed using probabilistic neural networks and neural classifiers in order to assign them to appropiate classes. The commands provide information about actions and objects. The data about tasks are based on operations and targets. The processes depend on parameters and working conditions. As the result, the extracted structured data with associated information allows for development of patterns and antipatterns for the crane control and safety analysis by uncovering implicit relations between commands, tasks and processes.

5 Conclusions and Perspectives

In the past few years, smart control systems have attracted a great deal of attention from both academia and industry due to many challenging research problems and a wide range of applications. The paper proposed an innovative concept of smart lifting devices with speech-based interaction using patterns and antipatterns generated with the use of a developed modeling methodology. It features reasoning models for inference mechanism of implicit relations between commands, tasks and processes in the interactive control. The developed methods can be used in automation of command recognition, safety analysis and assessment in human-machine interaction. Patterns and antipatterns generated from relational data mining used in the smart control process pose unique opportunities and challenges for interactive intelligence and its applications.

Acknowledgements. This project is financed by the National Centre for Research and Development, Poland (NCBiR), under the Applied Research Programme - Grant agreement No. PBS3/A6/28/2015.

References

1. Kumar, A., Metze, F., Kam, M.: Enabling the rapid development and adoption of speech-user interfaces. Computer **47**(1), 40–47 (2014). IEEE
2. Majewski, M., Kacalak, W.: Conceptual design of innovative speech interfaces with augmented reality and interactive systems for controlling loader cranes. In: Silhavy, R., Senkerik, R., Oplatkova, Z.K., Silhavy, P., Prokopova, Z. (eds.) Artificial Intelligence Perspectives in Intelligent Systems. AISC, vol. 464, pp. 237–247. Springer, Switzerland (2016). doi:10.1007/978-3-319-33625-1_22
3. Majewski, M., Kacalak, W.: Intelligent speech interaction of devices and human operators. In: Silhavy, R., Senkerik, R., Oplatkova, Z.K., Silhavy, P., Prokopova, Z. (eds.) Software Engineering Perspectives and Application in Intelligent Systems. AISC, vol. 465, pp. 471–482. Springer, Switzerland (2016). doi:10.1007/978-3-319-33622-0_42
4. Majewski, M., Kacalak, W.: Building innovative speech interfaces using patterns and antipatterns of commands for controlling loader cranes. In: International Conference on Computational Science and Computational Intelligence, pp. 525–530. IEEE Xplore Digital Library (2016)
5. Majewski, M., Zurada, J.M.: Sentence recognition using artificial neural networks. Knowl. Based Syst. **21**(7), 629–635 (2008). Elsevier
6. Specht, D.F.: Probabilistic neural networks. Neural Netw. **3**(1), 109–118 (1990). Elsevier

Application of Clustering Techniques for Video Summarization – An Empirical Study

Alvina Anna John[1(✉)], Binoy B. Nair[2], and P.N. Kumar[1]

[1] Department of Computer Science and Engineering,
Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham,
Amrita University, Coimbatore, India
cb.en.d.cse14008@cb.students.amrita.edu,
pn_kumar@cb.amrita.edu

[2] Department of Electronics and Communication Engineering,
Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham,
Amrita University, Coimbatore, India
b_binoy@cb.amrita.edu

Abstract. Identification of relevant frames from a video which can then be used as a summary of the video itself, is a challenging task. An attempt has been made in this study to empirically evaluate the effectiveness of data mining techniques in video summarization. Video Summarization systems based on histogram and entropy features extracted from three different color spaces: RGB, HSV and YC_BC_R and clustered using K-Means, FCM, GM and SOM were empirically evaluated on fifty video datasets from the VSUMM [1] database. Results indicate that clustering based video summarizations techniques can be effectively used for generating video summaries.

Keywords: Video summarization · Data mining · K-means · Self-Organizing Maps · Fuzzy C-means · Gaussian Mixture Models · Entropy

1 Introduction

Video summarization can effectively be used to convey the content in a video without the user having to go through the entire video [2, 3]. It is the process of extracting the important and relevant key frames that can represent the overall content of the video. Video summarization techniques help us to reduce the size of the video while preserving the semantics [4]. Avila et al. [5] presented VSUMM technique to produce the summaries. The technique implements a modified K-Means clustering technique for key frame extraction in HSV color model. Mahmoud et al. [6] presented the video summarization system VSCAN. It uses the HSV color space and modified DBSCAN clustering algorithm for obtaining the summaries. Wu et al. [7] proposed VRHDPS clustering algorithm for video summarization. Sachan et al. [4] worked on frameset based clustering to summarize the video. Ebrahim Asadi et al. [8] make use of HSV color space and Fuzzy C-means clustering technique for summarization of videos.

It is observed that there is scant literature on the effectiveness of different color spaces, feature selection techniques and clustering techniques on the video

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_49

summarization process. In the present study, an attempt has been made to address the issue by empirically evaluating twenty-four different video summarization models generated using a combination of the three color spaces: RGB, HSV and YC_BC_R; two feature election techniques: entropy and histogram and four clustering techniques: K-Means, Fuzzy C-Means (FCM), Self-Organizing Maps (SOM) and Gaussian Mixture Models (GM). All the summarization systems thus generated are empirically validated on fifty videos drawn from the VSUMM database [1]. Steps involved in the implementation of proposed summarization systems are presented in the following sections. Rest of the paper is organized as follows: Sect. 2 presents the system description, Sect. 3, presents the experimental results and the conclusions are presented in Sect. 4.

2 System Description

The proposed video summarization model is as shown in Fig. 1. Video data obtained is converted into frames and sampled. Features extraction techniques are then used to extract the feature vectors for images obtained after sampling. These feature vectors are then clustered. The first stage in the clustering process is to find the optimal number of clusters. Then, once the feature vectors are clustered for the optimal number of clusters, the vector closest to each centroid is identified and the image corresponding to this vector is considered to be the representative image for the whole cluster. The images thus selected are the output of the summarization technique.

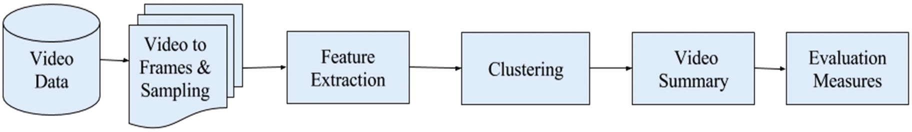


Fig. 1. Video summarization model

2.1 Color Spaces Considered

RGB. RGB color model is based on the Cartesian coordinate system. The R, G, B components in the RGB color system represents Red, Green and Blue respectively. The color images are usually stored in the RGB color space. The pixel values of R, G and B for an 8- bit image can range from 0 to 255.

$$[R]_{w\times h} + [G]_{w\times h} + [B]_{w\times h} = [Image]_{w\times h\times 3} \tag{1}$$

This is the color space in which the conventional video frames are recorded. In the present study too, all the videos considered are initially in RGB color space which are then converted into the required color space using transformation equations as described below.

HSV. HSV denotes the Hue, Saturation and Value. Hue denotes color and it is an angle ranging between 0 and 360°. Saturation gives information about the grey component and the range is between 0 and 100%. Value indicates brightness and has similar range like hue. The RGB to HSV conversion is given below [9].

$$H = \begin{cases} \theta & if\ B \leq G \\ 360 - \theta & if\ B > G \end{cases} \tag{2}$$

$$\text{Where} \quad \theta = \cos^{-1}\left\{ \frac{\frac{1}{2}[(R-G)+(R-B)]}{[(R-G)^2+(R-B)(G-B)]^{\frac{1}{2}}} \right\} \tag{3}$$

$$S = 1 - \frac{3}{(R+G+B)}[\min(R,G,B)] \tag{4}$$

$$V = \frac{1}{3}(R+G+B) \tag{5}$$

$\mathrm{YC_BC_R}$. $\mathrm{YC_BC_R}$ color model has one luminance component (Y) and two chrominance component ($\mathrm{C_B}$ and $\mathrm{C_R}$). The chrominance component deals with color information while the luminance component represents the light intensity. The formula to convert RGB to $\mathrm{YC_BC_R}$ is as given in equation below [10].

$$\mathrm{Y = 0.299R + 0.587G + 0.114B} \tag{6}$$

$$C_b = 128 - 0.168736R - 0.331264G + 0.5B \tag{7}$$

$$C_r = 128 + 0.5R - 0.418688G - 0.081312B \tag{8}$$

2.2 Feature Extraction Techniques

Once the video has been converted into frames in the required color space, the next step in summarization process is the extraction of relevant features from the frames. Two feature extraction techniques were considered: entropy based feature extraction and histogram based feature extraction. Each of the two techniques is briefly described as follows:

Entropy Based Feature Extraction. Entropy is an information based measure. The entropy of image is calculated using (9) [11].

$$Entropy = -\sum\nolimits_{i=1}^{n} p(x_i)\ .\ log_2 p(x_i) \tag{9}$$

In all the color spaces considered here, the image can be represented as a three dimensional matrix of the type $w \times h \times 3$ where w and h represent the image dimensions and 3 is the number of components (channels) in the color space. Entropy of each component is calculated and put together to form the feature vector. Assuming there are N images, there will be N × 3 feature vectors.

Histogram Based Feature Extraction. Histogram gives information about the distribution of color in an image. The histogram of each channel of a frame is calculated with bin size as 16 [5]. Then, these histograms are put together to form the feature vector. If there are N images after the histogram based feature extraction, the size of the resultant dataset will be N × 48.

2.3 Clustering Techniques Evaluated

Once the feature vectors are obtained, the next step is to group similar images. Machine learning based systems have been found to be well capable of handling complicated data patterns for example in [12]. In this paper, clustering algorithms are used to group similar images. Effectiveness of four clustering algorithms, namely, K-Means, FCM, GM and SOM was evaluated. The four algorithms are explained briefly below.

K-Means [13]. K-means is a partitional clustering algorithm. The number of clusters (k) is specified at the outset. Each data point can be represented as an m-dimensional vector in data matrix X with the number of data points being denoted by N. Hence, $\boldsymbol{X} = \begin{bmatrix} x_{11} & x_{12} & \ldots & x_{1M} \\ x_{21} & x_{22} & \ldots & x_{2M} \\ \vdots & \vdots & \vdots & \vdots \\ x_{N1} & x_{N2} & \ldots & x_{NM} \end{bmatrix}$. Initial cluster centroids $\boldsymbol{C_j} = \{C_1, C_2, \ldots, C_k\}$ are selected using K-means++. The distance between the data points and all the cluster centroids are calculated using Euclidean distance metric. The data point will be clustered into the cluster with the nearest centroid. New centroid is calculated for each cluster after new data point is added to it. The process of calculating the distance and new centroids is repeated until no further reallocation of data points happen or after maximum of 100 iterations.

Fuzzy C-Means (FCM) [14]. In FCM clustering, the data points can belong to multiple clusters, but with varying degrees of membership. These membership values are used to decide the degree of association of the data points with the clusters. Higher values depict large degree of association of the data point with that particular cluster [8]. Given n data points with l features and k number of centroids, $\boldsymbol{C_j} = \{C_1, C_2, .., C_k\}$ and partition matrix $W = w_{ij} \in \; [0,1], i = 1,2,\ldots n, \; j = 1,2,3\ldots k,$ the objective is to minimize the objective function (10). The level of cluster fuzziness is denoted by m. The algorithm is repeated until convergence i.e. until it reaches the sensitivity threshold $\varepsilon = 10^{-3}$

$$\arg\ \min_k \sum\nolimits_{i=1}^{n} \sum\nolimits_{j=1}^{k} w_{ij}^{m} \left\| x_i - c_j \right\|^2 \tag{10}$$

where,

$$w_{ij} = \frac{1}{\sum\nolimits_{p=1}^{k} \left(\frac{\left\| x_i - c_j \right\|}{\left\| x_i - c_p \right\|} \right)^{\frac{2}{m} - 1}} \tag{11}$$

$$c_p = \frac{\sum\nolimits_x w_p(x)^m x}{\sum\nolimits_x w_p(x)^m} \tag{12}$$

Self–Organizing Maps (SOM). Self–Organizing Maps are a type of Artificial Neural Network (ANN) that can be used for clustering of data. SOMs have been shown to be capable of successfully handling non-linear datasets [15]. SOM cluster the data points based on the concept of similarity [16]. First the weights of the nodes are initialized. The similarity between the input vector chosen and weight vectors is found. The node whose weight vector is most similar to input vector is called the winning node or the Best Matching Unit(BMU). The nodes in the neighborhoods of BMU are updated [16].

Gaussian Mixture (GM) Based Clustering. GM models can be effectively used for clustering of data points [17]. The number of clusters must be specified by the user and it denotes the number of GM model components used. The GM model is fitted using the Expectation-Maximization(EM) algorithm. In the Expectation step, the probability that data point belongs to a cluster is calculated and based on this probability, recalculation of cluster means and covariances are performed in the Maximization step. The probability density function of Gaussian is as shown in (13) [18]. The GM models perform clustering by allocating the data points to the components having the highest posterior probability. GM model is most commonly used for background subtraction in many applications to detect foreground objects [19].

$$g_j(X) = \frac{1}{\sqrt{(2\pi)^n \left| \sum\nolimits_j \right|}} e^{\frac{-(X-\mu_j)^T \sum\nolimits_j^{-1} (X-\mu_j)}{2}} \tag{13}$$

where,
$g_j(X)$ is Probability Density Function of Gaussian
j is the cluster number, j = 1,2…,k
$\sum_j$ is the covariance matrix for cluster j
μ_j is the mean of cluster j.

2.4 Evaluation Metrics

Optimal Cluster Evaluation Metric [20]. The identification of optimal number of clusters is performed using the Calinski-Harabasz criterion, also called as the Variance Ratio Criterion (VRC) as in (14).

$$VRC_k = \frac{SS_B}{SS_W} \times \frac{(N-k)}{(k-1)} \tag{14}$$

SS_B and SS_W denote the inter cluster and intra cluster variance respectively, while N denotes the number of observations. k signifies the cluster size and the value of k ranges from four to twenty. The k value for which we obtain the highest Calinski-Harabasz index value is the optimal number of clusters.

Summary Evaluation Metric. The metric used for evaluation of the summarization effectiveness is Comparison of User Summaries–Accuracy Rate ($CUS_{A)}$ [5] and is calculated by comparing the User Summary (US) and Automatic Summary(AS). AS is generated using the summarization method. Equation (15) defines CUS_A. The worst case and best case value is 0 and 1 respectively for CUS_A. The worst case occurs when none of the frames of AS matches with US while best case scenario is when all the frames of AS matches with US.

$$CUS_A = \frac{n_{mAS}}{n_{US}} \tag{15}$$

where,

n_{mAS} = number of matched frames.
n_{US} = total number of user summary frames.

3 Experiments and Results

Fifty videos, belonging to different genres in the Open Video Project were taken for experimentation from the VSUMM database [1]. All the videos were in MPEG-1 format with frame rate of 30 fps. Sampling rate is taken as one frame per second, as in [5]. All possible combinations of color spaces, feature selection techniques and clustering algorithms were considered, resulting in twenty-four different video summarization systems. The process is presented in Fig. 2 below.

The performance of each of these systems is presented in figures (Figs. 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13 and 14). In all the histograms below, the X- axis of the histogram plot represents CUS_A while the Y-axis denotes the number of videos in the bin (all

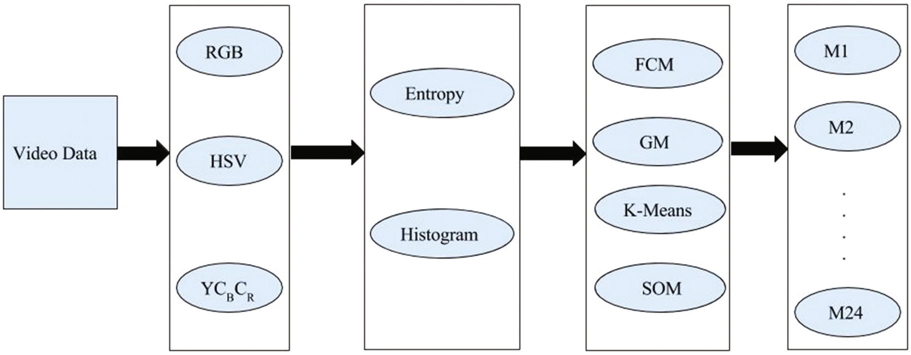


Fig. 2. Proposed model

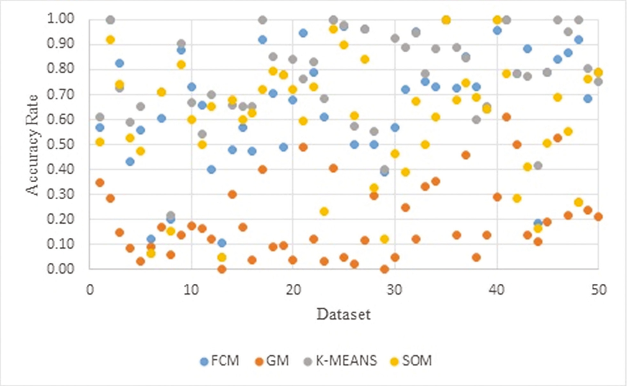


Fig. 3. CUS$_A$ Scatter Plot for RGB-Entropy-Clustering based summarization systems

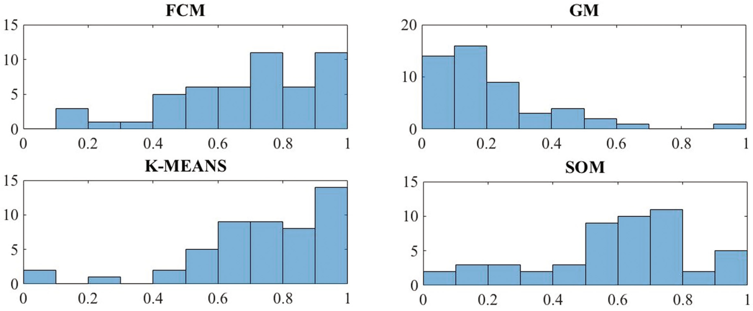


Fig. 4. CUS$_A$ Histogram Plot for RGB-Entropy-Clustering based summarization systems

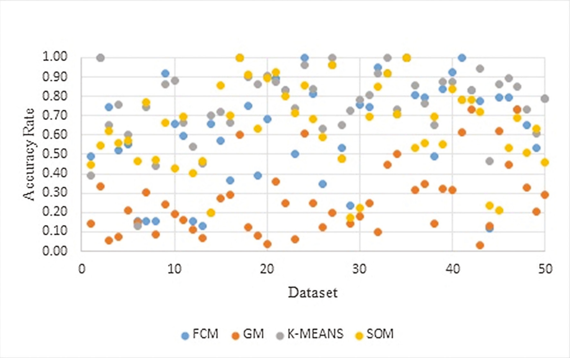


Fig. 5. CUS$_A$ Scatter Plot for HSV-Entropy-Clustering based summarization systems

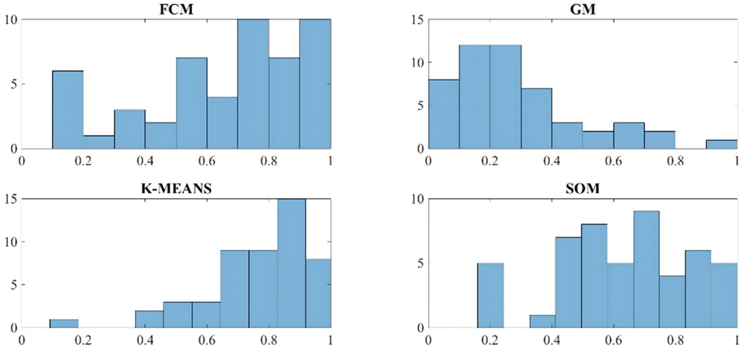


Fig. 6. CUSA Histogram Plot for HSV-Entropy-Clustering based summarization systems

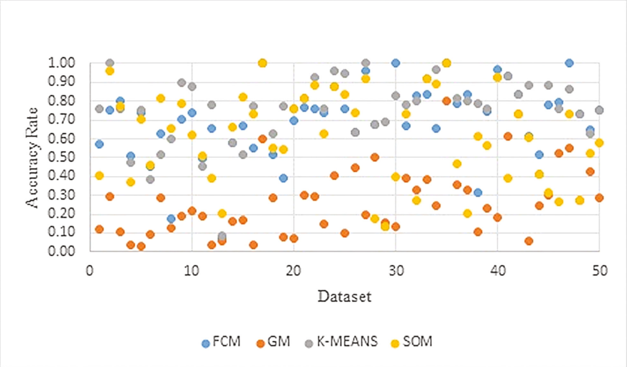


Fig. 7. CUS$_A$ Scatter Plot for YC$_B$C$_R$-Entropy-Clustering based summarization systems

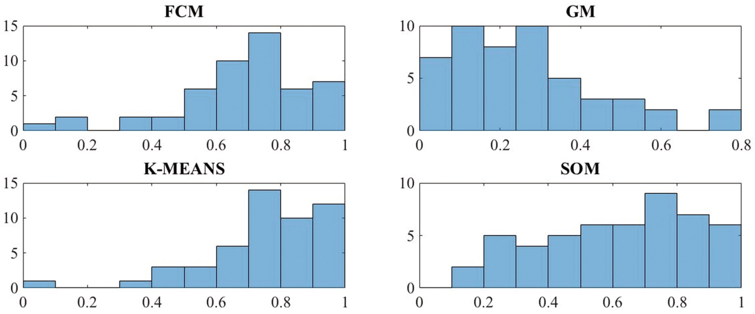


Fig. 8. CUS$_A$ Histogram Plot for YC$_B$C$_R$-Entropy-Clustering based summarization systems

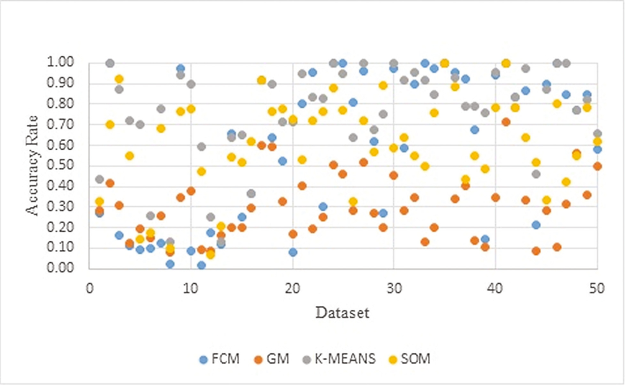


Fig. 9. CUS$_A$ Scatter Plot for RGB-Histogram-Clustering based summarization systems

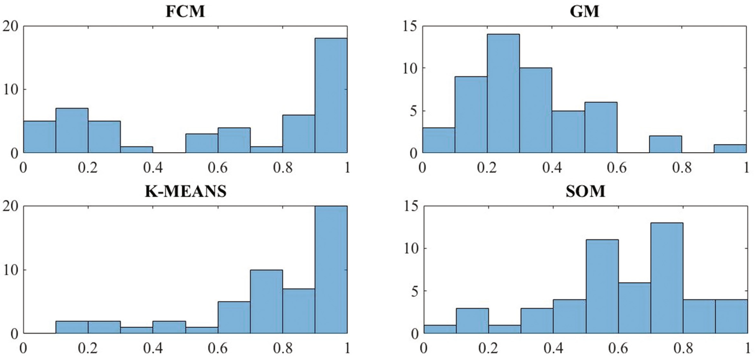


Fig. 10. CUS$_A$ Histogram Plot for RGB-Histogram-Clustering based Summarization Systems

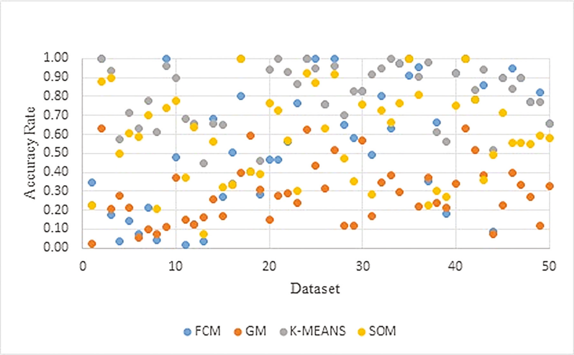


Fig. 11. CUS$_A$ Scatter Plot for HSV-Histogram-Clustering based summarization systems

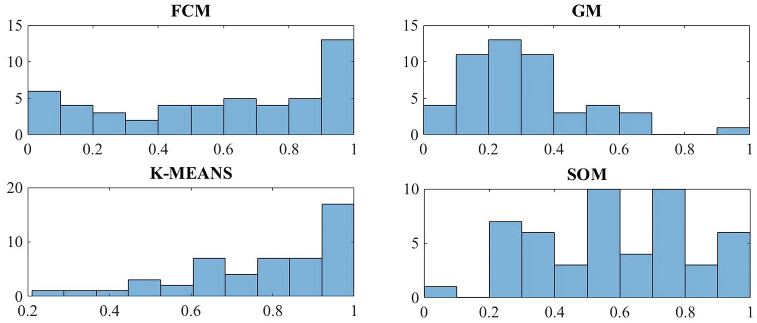


Fig. 12. CUS$_A$ Histogram Plot for HSV-Histogram-Clustering based summarization systems

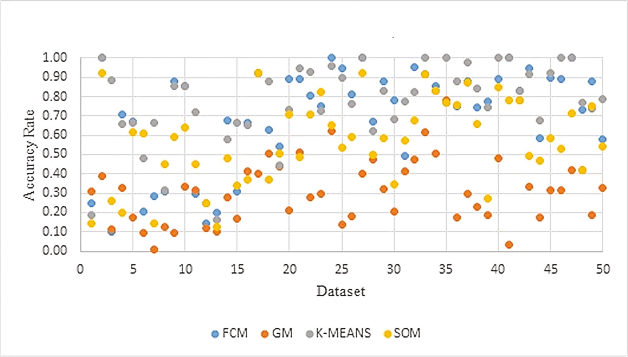


Fig. 13. CUS$_A$ Scatter Plot for YC$_B$C$_R$-Histogram-Clustering based summarization systems

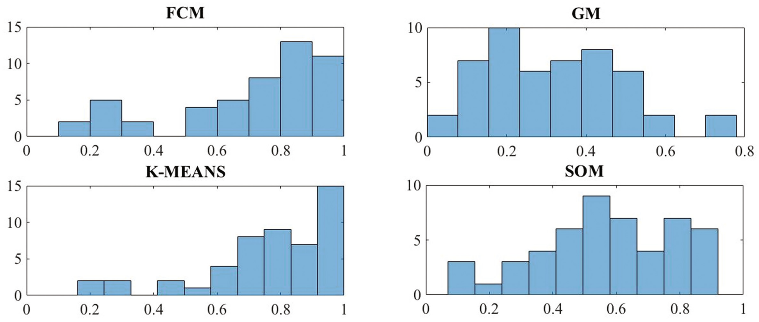


Fig. 14. CUS$_A$ Histogram Plot for YC$_B$C$_R$-Histogram-Clustering based summarization systems

histograms employ equal width binning with bin numbers = 10). The scatter plots given below indicate the CUS$_A$ values for the individual videos considered in the VSUMM dataset and each color indicates the clustering technique involved.

Figures 3 and 4 show the performance of models M1(RGB-Entropy-FCM), M2 (RGB-Entropy-GM), M3(RGB-Entropy-K-Means) and M4(RGB-Entropy-SOM). It can be observed that K-Means exhibited the best performance followed by FCM and SOM, respectively. Worst performance was shown by GM clustering based systems.

Figures 5 and 6 show the performance of models M5(HSV-Entropy-FCM), M6 (HSV-Entropy-GM), M7(HSV-Entropy-K-Means) and M8(HSV-Entropy-SOM). It can be observed that K-Means showed better performance than FCM and SOM in this case as well. The GM clustering based systems exhibited worst performance.

Figures 7 and 8 show the performance of models M9(YC$_B$C$_R$-Entropy-FCM), M10 (YC$_B$C$_R$-Entropy-GM), M11(YC$_B$C$_R$-Entropy-K-Means) and M12(YC$_B$C$_R$-Entropy-SOM). K-Means based systems outperformed all other systems in this case as well. GM clustering based systems showed poor performance in this case as well.

Figures 9 and 10 show the performance of models M13(RGB-Histogram-FCM), M14(RGB-Histogram-GM), M15(RGB-Histogram-K-Means) and M16(RGB-Histogram-SOM). It can be observed that K-Means based systems exhibited better performance than FCM and SOM. Worst performance was shown by GM clustering based systems.

Figures 11 and 12 show the performance of models M17(HSV-Histogram-FCM), M18(HSV-Histogram-GM), M19(HSV-Histogram-K-Means) and M20(HSV–Histogram-SOM). It can be noticed that K-Means showed better performance than FCM and SOM. GM clustering based systems exhibited worst performance.

Figures 13 and 14 show the performance of models M21(YC$_B$C$_R$-Histogram-FCM), M22(YC$_B$C$_R$-Histogram-GM), M23(YC$_B$C$_R$-Histogram-K-Means) and M24 (YC$_B$C$_R$ -Histogram-SOM). It can be seen that K-Means showed the best performance, followed by FCM and SOM. GM clustering based systems showed worst performance in this as well.

It can be seen from the results summarized in Figs. 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13 and 14 that K-means clustering based video summarization systems are able to outperform other summarization models considered. The best video summarization performance was obtained by the model M19(HSV-Histogram-K-Means).

4 Conclusions

In this paper, we presented 24 different models for video summarization, formed by the combinations of color models, feature vectors and clustering algorithms. Video summarization technique was effectively able to reduce the size and provide efficient summary. We have used the accuracy rate measure CUS_A to evaluate the models. The K-Means implemented in a HSV color space with histogram feature extraction technique provided better results than the other clustering algorithms. Of the other techniques considered, FCM showed slightly better performance than SOM clustering techniques for all color spaces using the entropy features and considerably better results in the YC_BC_R color space for histogram features. SOM algorithm showed the highest accuracy rate using the HSV-Entropy model. GM clustering exhibited poor performance among all the algorithms.

References

1. VSUMM Database. https://sites.google.com/site/vsummsite/download
2. Sebastian, T.: A survey on video summarization techniques. Int. J. Comput. Appl. (0975–8887) **132**(13), 31–33 (2015)
3. Geetha, S.P., Thiruchadai Pandeeswari, S.: Visual attention based keyframes extraction and video summarization. In: CS IT-CSCP 2012, vol. 2, pp. 179–190 (2012)
4. Sachan, P.R., Keshaveni: Frame clustering technique towards single video summarization. In: Second International Conference on Cognitive Computing and Information Processing (CCIP) Frame, pp. 1–5 (2016)
5. De Avila, S.E.F., Lopes, A.P.B., Da Luz, A., De Albuquerque Araújo, A.: VSUMM: a mechanism designed to produce static video summaries and a novel evaluation method. Pattern Recogn. Lett. **32**(1), 56–68 (2011)
6. Mahmoud, Karim M., Ismail, Mohamed A., Ghanem, Nagia M.: VSCAN: an enhanced video summarization using density-based spatial clustering. In: Petrosino, A. (ed.) ICIAP 2013. LNCS, vol. 8156, pp. 733–742. Springer, Heidelberg (2013). doi:10.1007/978-3-642-41181-6_74
7. Wu, J., Zhong, S., Jiang, J.: A novel clustering method for static video summarization. Multimed. Tools Appl. pp. 1–17 (2016)
8. Asadi, E., Charkari, N.M.: Video summarization using fuzzy C-Means clustering. In: 20th Iranian Conference on Electrical Engineering, pp. 690–694 (2012)
9. Gonzalez, R.C., Woods, R.E.: RGB to HSV. Digital Image Processing (1992)
10. YC_BC_R. http://www.roman10.net/2011/08/18/ycbcr-color-spacean-intro-and-its-applications/
11. Phan, R.: Entropy. https://www.codementor.io/tips/9423772841/how-to-calculate-the-shannon-entropy-of-a-part-of-image-data
12. Balaji, D.A.J., Ram, D.S.H., Nair, B.B.: Modeling of consumption data for forecasting in automated metering infrastructure (AMI) systems. In: Proceedings of the 5th Computer Science On-line Conference (CSOC 2016), vol. 3, pp. 165–174 (2016)

13. Matlab, K Means Clustering. https://in.mathworks.com/help/stats/kmeans.html
14. Wikipedia, Fuzzy C Means. https://en.wikipedia.org/wiki/Fuzzy_clustering
15. Nair, B.B., Kumar, P.K.S., Sakthivel, N.R., Vipin, U.: Clustering stock price time series data to generate stock trading recommendations: an empirical study. Expert Syst. Appl. **70**, 20–36 (2017)
16. Arumugadevi, S., Seenivasagam, V.: Comparison of clustering methods for segmenting color images. Indian J. Sci. Technol. **8**(7), 670–677 (2015)
17. Matlab, GMMClustering. https://in.mathworks.com/help/stats/clustering-using-gaussian-mixture-models.html
18. McCormick, C.: GMMClustering. http://mccormickml.com/2014/08/04/gaussian-mixture-models-tutorial-and-matlab-code/
19. Ou, S., Lee, C.: Low Complexity on-line video summarization with gaussian mixture model based clustering. In: IEEE International Conference on Acoustic, Speech and Signal Processing, pp. 1260–1264 (2014)
20. Matlab, Optimal Cluster Evaluation. https://in.mathworks.com/help/stats/clustering.evaluation.calinskiharabaszevaluation-class.html

Survey on Various Traffic Monitoring and Reasoning Techniques

H. Haritha(✉) and T. Senthil Kumar

Department of Computer Science and Engineering,
Amrita School of Engineering, Amrita Vishwa Vidyapeetham,
Amrita University, Coimbatore, India
cb.en.d.cse14010@cb.students.amrita.edu,
t_senthilkumar@cb.amrita.edu

Abstract. Traffic monitoring and surveillance is advancing in recent years. This paper proposes a survey on an overview of various traffic sensing, monitoring techniques. Several projects have been developed for the detection and tracking of vehicles on multiple scenarios. The vehicle monitoring results depends mainly on the camera positioning. This paper gives a detailed description on the different camera positioning and monitoring which includes straight roads and intersections. In this survey a detailed description is given on preprocessing techniques, vehicle detection and tracking methods. Finally paper concludes with the challenges and the future scope.

Keywords: Vehicle surveillance · Vehicle sensing · Monitoring · Vehicle tracking · Pre-processing · Shadow detection · Shadow elimination · Occlusions

1 Introduction

Computer vision technology has been improving rapidly in the past few years. One of its application around the world is vision based traffic monitoring. As the road traffic is being increasing day by day, our concern for the safety too is increased. Several Projects have been launched from the earlier 1990's on the vehicle monitoring and controlling. Still unfortunately many video surveillance systems existing are depending their works on humans. Vision based traffic monitoring has a wide range of applications in day to day life. Many systems have already been developed for the detection and tracking. Still the existing many vision based algorithms are not robust enough to do the automatic detection of these events [6]. The computer vision technology involves acquiring, processing, analyzing and understanding of the digital data acquired from the video cameras. Vision based traffic monitoring has a wide range of applications in a day to day life. Abnormal event detection is a major application, in which it involves accident detection, traffic violation, parking violation, zig zag driving etc. The main goal of the vision based traffic monitoring system is to act as a human eye that should detect the abnormal activities in road traffic and to give alert without any delays. Providing intelligence to the system could achieve this result, in spite of the huge amount of data and heavy processing (Fig. 1).

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_50

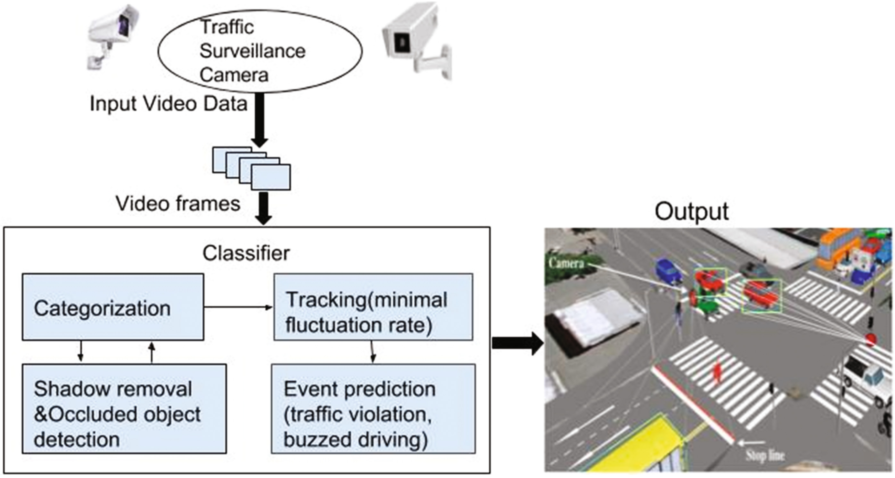

Traffic Surveillance Camera
Input Video Data
Video frames
Classifier
Categorization
Tracking(minimal fluctuation rate)
Shadow removal &Occluded object detection
Event prediction (traffic violation, buzzed driving)
Output
Camera
Stop line

Fig. 1. Architecture diagram of vehicle tracking.

Our paper is organized as follows, Sect. 2 includes various vehicle sensing techniques till the recent technologies, then Sect. 2 gives a brief overview on various camera positioning and calibration techniques, Sect. 3 and 3.1 includes various traffic analysis and vehicle monitoring techniques on both straight road and on intersections, Sect. 3.2 gives a detailed description and various methods of pre processing. Section 3.3 includes the various vehicle detection counting and tracking algorithms. Section 3 represents the occlusion management technique available till recent. Section 3.4 gives a comparative study on the various techniques available for the vehicle tracking techniques in a detailed table (Table 1).

2 Vehicle Sensing Techniques

2.1 Infrared Detectors

There are two type of infrared detectors active and passive. They can actively work at both day and night times and detects vehicle position, class, count and speed. The advantages of these infrared detectors is that they can be easily mounted on the road sides and also multiple lane functioning can also be done, but the disadvantage is that they are very sensitive to extreme weather conditions like rain, blowing snow or snow where the visibility is less than 20 feet [11].

2.2 Microwave/Millimeter Wave Radar

Microwave detectors detect the presence, speed and velocity of a moving vehicle [9]. Operate by measuring the energy reflected by the target vehicle, in the field of view. Multi lane operation is available and also resistant to difficult weather conditions. The improper placement of the detectors will also affect the accuracy in the output.

2.3 Acoustic Detectors

Array of acoustic detectors are helpful in determining the presence of a moving vehicle. Advantage of these detectors are they are multilane functional and also passive [11]. But the slow moving vehicles and certain model vehicles are not detected [11]. Extreme cold temperature will also affect the output rate.

2.4 Ultrasonic Detectors

Ultrasonic detectors generates the high frequency sound waves that hits the target and measures the energy reflected from object by calculating the distance between the sending and receiving wave. It should be placed perpendicular to the target hence it could not detect the occluded vehicles [10].

2.5 Inductive Loop Detectors

Inductive loop detectors are the electrical conducting loop which are insulated that is embedded in the pavement [13]. They can detect the moving vehicles passing through the top of it. They are the widely used technology for vehicle detection in the United States [9]. They detects the speed, presence, gap and headway but the installation and maintenance require the cutting of pavement which will obviously decrease the life of the pavement [11].

2.6 Magnetic Detectors

Magnetic detectors are used for detecting the presence and the movement of vehicles. They are of active and passive type. Active type detects both the presence and the movement of the vehicle but the passive type detects only the movement. They can be used where the loops cannot be installed [11]. They work by producing a magnetic field and when a vehicle moves it cuts the magnetic field and the vehicle movement is detected. The advantage is that they can withstand the pressure caused by the traffic more than the loop systems and also they are resistant to extreme weather conditions [11].

2.7 Piezoelectric Detectors

They can get the presence of any vehicle that is being stopped with on it. They convert the pressure, force, acceleration, strain to electric energy using the piezoelectric effect. When a vehicle passes over it the piezoelectric material is being compressed and the voltage is produced. Which in turn detects the presence of the vehicle. It has the advantage of detecting the exact vehicle position and also the path of the vehicle movement. Vehicle speed can also be detected by placing the two of them in series [9]. The disadvantage is that the road has to be cut every time for the installation, and for the maintenance.

2.8 Acceleration Detectors

Acceleration detectors are used for traffic monitoring and vehicle detection. The system operates for a long period of time freestanding. The advantages of detecting the acceleration is that the movement of the vehicle can be obtained and hence can detect the vehicles that are moving against the traffic rules like increase speed, red light stopping, no left turns etc. [14]. They can be used to determine the car crashes, which is being applied in many car manufacturing companies.

2.9 Spread Spectrum Wideband Radar

It is an accurate range detector [9]. They are used for detecting the motion of vehicle. This sensor system has wide range of signals and it starts by looking at a single range and while the vehicle starts moving the range fluctuates, and hence the vehicle motion is being detected. The advantages of spread spectrum wideband radar is that the result is very accurate and occlusions are not a problem. We can get the additional information from the traffic [10] and also they are very cheap. The disadvantages involved in here is that they are single lane detectors.

2.10 Video Detectors

Video image detectors are widely being used. They have a wide range of data collection [11]. They can be installed easily on the road side or on the top bridges facing the front of the vehicle as well as the rear side. They can be used to detect the vehicle presence, vehicle movement and also the other information related to traffic flow. There are various advanced algorithms for processing the output from the video camera. They can be easily used for changing or adding new detection zones [11].

3 Traffic Analysis Methods

3.1 Vehicle Monitoring (Intersection and Straight Road)

Transportation has been increasing rapidly day by day and vehicles are all around. The monitoring of these vehicles are being done pretty easily from point to point with the emerging technologies around. The road types for monitoring has been divided to straight roads, intersections and curved roads. There are various algorithms developed for the monitoring in these areas. Many of the vehicle monitoring applications include getting the vehicle count, vehicle path, flow rates, density of vehicle, weight and length of the vehicle, class differentiation of the vehicle and the identity of the vehicle [3]. The monitoring in urban and highways vary in many perspectives. The urban area have low camera angle and high density of vehicles, hence occlusion is a main problem. Freeways are having wide angle camera focus and also homogenous vehicle flow [8].

3.2 Preprocessing Techniques

Video Input data is divided in to frames and then dispatched for preproessing. Preprocessing aims at the enhancement of image data, improvement of the features for the further processing and to exclude the misrepresentations. Image preprocessing uses the repetition in images for the processing [17]. Preprocessing is performed before feature extraction which corrects various fragments in the image that includes lighting changes [12], sensor variations, noise, geometric corrections, color corrections [16]. Image enhancement is another result of preprocessing which includes, sharpening, color balancing, scale-space pyramids, illuminations, blur and focus enhancement. The basic preprocessing techniques includes:

Image resizing

Image resizing is one of the preprocessing methods. Nearest neighbor interpolation method is one of it where the value of any non given point in a space is approximated. Seam carving is another image resizing method without the geometric constraints [18]. There by the image can be processed with minimum time consumption and occupying less memory space.

Image Normalization

Normalization is another preprocessing method. In an image the distribution of gray level/intensity varies. Also illumination variation contains non uniform contrast. Image normalization acts in the above mentioned scenarios. The pixel intensity values are affected by the normalization. Histogram normalization is one of the commonly used image normalization technique. The other various normalization techniques includes zero padding, track extension, resampling, smoothing, thereby can control the variation in the intensity of pixel values.

Image denoising/filtering and cropping

Gaussian noise is the default noise that occurs in an image while its been captured. The image denoising regains the image without the noises. Image smoothing is the widely used approach from the older days for the removal of noises. In traffic data the noises are more from weather conditions like fog, mist, rain etc. Level set approach is one of the gratifying noise removal approach, where the pixels are viewed as topographic maps. Another approach is motion of curvature, where the spikes of noises will disappear quickly. The advantage is that the boundaries will remain still sharp. The basic division of filtering methods include transform domain filters and spatial domain filtering methods [18].

Dimensionality reduction

Dimensionality reduction concerns with feature vectors. Down sampling is one of the method for dimensionality reduction, where taking the average number of block of pixels in a regular grid and removing the other pixels. Dimensionality reduction can be divided into feature selection and feature extraction. The other dimensionality reduction techniques include vector quantization, polynomial fitting, multi resolution decomposition, Hidden markov model, spectral methods, kernel methods, Principal component analysis, Kernel Principal component Analysis, Graph based kernel PCA, Linear Discriminant Analysis, Generalized discriminant Analysis. Hence from as a result of this technique by averaging the values the most relevant datas are included for the further processing without the whole bulky data.

Brightness thresholding
Simple and effective way preprocessing method by dividing image into background and foreground. It works by converting grayscale image to binary image. By the application of brightness threshold the image can be distinguished into object of interest and the other part. Here the threshold value is set and the pixel value is calculated from the given image, if the pixel values is brighter than the threshold value then convert it to black or white depending on the condition. One of the disadvantage involved in thresholding is that the pixel intensity values are only considered and hence in case of pixels related to the neighbors are separated.

3.3 Vehicle Detection

After the preprocessing the particular target detection can be performed. Detection implicates locating vehicles on video frames. Vehicle detection plays a crucial role in Intelligent Transportation System which results in making a better system that acts more smarter, safer and integrated. Vehicle detection using video camera is a non intrusive form of detection which mainly involves motion segmentation and environment/background modeling [4]. Other processes like classification of vehicles, tracking vehicles and their behavior recognition depends on this. The vision-based vehicle detection which includes vehicle candidate localization, that involves more number of frames and the methods are classified into background subtraction, model based segmentation, feature based segmentation, motion-based segmentation, frame differencing [1]. Vehicle detection using multi cameras (stereo vision) as well as single camera where they are placed over the road and fixed. The system gives 90% accuracy with the stereo vision. Multi feature detection and 3D tracking methods are employed for the detection where the maximum spanning tree clustering algorithm is applied [19]. Various sturdy algorithms are developed for the detection of vehicles which resulted in less than 9% error rate.

Classifier algorithms acts smartly on the detection cases. The classifiers based on Histogram of Oriented Gradient (HOG) and Gabor have attained a better performance in accuracy rate and performance than Principal Component Analysis which is another classifier and others based on gradient and symmetry, but study shows that they have limitations under certain scenarios. The feature combination results better since Gabor and HOG fails in far range and the latter in middle range [5]. There are other detecting algorithms on the basis where detection is based on points. Point detectors are detectors that locates the interesting points in image which includes KLT detector, SIFT detector, Harris detector, Moravec's detector. With the supervised learning mechanism the detection can be performed automatically from different object views [6].

Shadow detection
For the proper detection of targets shadows in target objects should be detected and removed. Shadows results in serious problems in the processing of images. Shadows are falsely taken as foreground objects, they causes shape distortion in images,

produces error in vehicle counting and they also acts as occlusions [15]. Shadows are detected with many features like chromaticity, physical properties, intensity, textures, geometry, and temporal features. The various physical methods include kernel based, semi supervised, Gaussian mixture model, kernel based and various texture based methods like gabor filter, ratio-edge test, Principal Component Analysis (PCA) based, Gradient background subtraction, local ternary pattern MRF etc.

Line-based shadow algorithm is another approach proposed by Jun-Wei et al. [20] in automatic traffic surveillance system where the algorithm uses a group of lines to eliminate all the nonessential shadows. Automatic lane-dividing lines are proposed with the experimental results that showed the system is more powerful, accurate, and fast. Haar-like feature with Adaboost is an approach for the shadow detection where the Haar detector can be trained offline. Shadows are easily extracted using visual data.

3.4 Vehicle Tracking

Traffic surveillance includes a lot of functions like vehicle detection as well as tracking. Vehicle tracking is the important processing stage in surveillance. Tracking means trailing a particular target. There are a lot of advantages in tracking vehicles in traffic scenes like safety, security, reducing the crimes, and apart from these we can also keep track of the flow of vehicles and velocity. A lot of algorithms have been developed in the recent years for the efficient tracking. Brendan et al. [2] in their paper created a scene modeling for tracking, where the object is to be identified and the identity should be maintained in each frame. The tracking states are $S_T = \{S_1, S_2, \ldots, S_T\}$, from S_T portray the position, appearance, shape, velocity and the other object labels. One of the tracking algorithm is extended lucas kanade template matching algorithm with a speed accuracy of 2.3% for 95% of the vehicles in the varying lighting conditions [7]. False detection is an issue that highly affects tracking, temporal tracking method overcomes this false detection problem. Tracking should works effectively on the moving scenes huddles, numerous position changing objects, varying illumination conditions and the other different changes in the target sceneries. The tracking methods are categorized on the basis of:

Region based method
Here the parameters are regions of the data. In this method the ith frame and the i + 1th frame in together produce a conflict free association graph. They are sensitive to partial occlusions. The spatio-temporal information can also be used for deducing the regions. Region based tracking method handles only the object-level entities.

Model-based tracking method
This is used for the free-flowing traffic. A 3D model of object is created, and the result is highly accurate. They are used for classifying and tracking the moving objects in a crowded and undisciplined landscape. The model dwells with 3D geometrical portrayal of vehicles. Both the texture and edge information can be used as models. M-estimators are also included to impose shadows, occlusions and varying backgrounds which may result in false detection.

Contour and feature based tracking method

A rough contour of the object is gleaned from the first frame. The further contours can be created using kalman filter. The acceleration and velocity of the object can be obtained by this tracking method. This gave more robustness and could handle changing behaviors and varying lighting conditions. Tracking by extracting the features of the target objects is a form of unsupervised object tracking method.

Point tracking and silhouette tracking methods

Point tracking is another tracking method for shorter distance. It uses Kanade-Lucas-Tomasi (KLT) feature tracking algorithm. Here the objects are represented as points and tracked crosswise the frames. Silhouette tracking is another method that tracks objects that are in solid form. Silhouette extraction methods are available from earlier times like the subtraction method, background modeling etc. The other methods include parmish silhouette approach, silhouette mapping, spatio-temporal shape extraction from silhouette tracking. These methods are applied for objects with constant shape throughout.

Table 1. Comparison of recent tracking methods

Algorithm	Technique	Advantage	Shadow removal	Occluded Object Detection
Symmetric frame differencing target detection algorithm	Based on local clustering segmentation	Target detection and tracking	No	Partial
Mean shift tracking	Density estimation	Detect occluded targets	No	Partial
Contourlet transform tracking	spectrum data estimation	Real time tracking accuracy	No	No
Block-matching algorithm	Based on motion estimation	Reduced computation time	No	No
Compressive tracking algorithm	Application of kalman filter	High robustness and real time tracking	No	Partial
Optical Flow	Blob Analysis Method	Detect, track and count objects in real time	No	No
Markov chain monte carlo data association	Approximation of optimal bayesian filter	Multiple object tracking in dense scenes	Yes	No
Optimal unbiased finite memory filter	Adaboost using proposed feature	Outperforms kalman and other filters	No	Yes
K-shortest disjoint path	motion - based optimization on dual graphs	Accuracy rate of 99.4%	Yes	Yes
Kanade-Lucas-Tomasi	Applied to motion layers	High robustness	Yes	Yes

Kernel tracking
Kernel based tracking improves the robustness and accuracy of tracking. One of this is central component method in which the centroids of all the objects are taken for tracking multiple objects in single frame. In case of moving scenes Isotropic kernel-method is applied that regularizes the feature histogram based target representations. They successfully works with the moving scenes, occlusions, eloquent cameras, scale variations etc.

4 Challenges and Future Work

Vehicle surveillance is still a research area and lot of challenges arising. The weather condition and illumination changes are the major challenges that we are facing now a days. The fog, mist, rain or poor lighting conditions could make the view from the camera very poor and hence the tracking and detection can be disturbed. The occlusion is another major problem in tracking. The target vehicles may be partially or fully occluded which is very tough to be detected. In the case of reconstructions, 3D reconstructions has not yet achieved from the intersections. The vehicle motion pattern learning, trajectory prediction and vehicle tracking accuracy is still remaining as a challenge.

5 Conclusion

Video surveillance is an important topic in computer vision. It has got a lot of applications for the peoples in daily activities. Here the paper presents a literature survey on the complete vehicle surveillance top to bottom. Paper gives a brief review on the various vehicle sensing techniques from the earlier days. The input for the processing is the video data taken from the camera which is positioned in various places of the road. Hence the camera positioning plays a crucial role in the surveillance. This survey gives a detailed description on the various preprocessing techniques implemented so far and then the various detection and vehicle tracking techniques available. Accuracy of various implemented algorithms implementation results have been mentioned. Future works and challenges helps in gaining more knowledge and an encouragement for the future research.

References

1. Sanchez, A., Suarez, P.D., Conci, A., Nunes, E.: Video-based distance traffic analysis: application to vehicle tracking and counting. Comput. Sci. Eng. **13**(3), 38–45 (2011)
2. Morris, B.T., Trivedi, M.M.: A survey of vision-based trajectory learning and analysis for surveillance. IEEE Trans. Circ. Syst. Video Technol. **18**(8), 1114–1127 (2008)
3. Setchell, C.J.: Applications of computer vision to road-traffic monitoring. Technical report, University of Bristol, Bristol, UK © (1997)

4. Ko, T.: A survey on behavior analysis in video surveillance for homeland security applications. In: 37th IEEE Applied Imagery Pattern Recognition Workshop, AIPR 2008 (2008)
5. Arrospide, J., Salgadoe, L.: A study of feature combination for vehicle detection based on image processing. Sci. World J. **2014**, 13 (2014). doi:10.1155/2014/196251. Article ID 196251
6. Rout, R.K.: A Survey on Object Detection and Tracking Algorithms, Department of Computer Science and Engineering National Institute of Technology Rourkela Rourkela – 769 008, India
7. Loureiro, P.F.Q., Rossetti, R.J.F., Braga, R.A.M.: Video processing techniques for traffic information acquisition using uncontrolled video streams. In: 12th International IEEE Conference on Intelligent Transportation Systems, ITSC 2009 (2009)
8. Buch, N., Velastin, S.A., Orwell, J.: A review of computer vision techniques for the analysis of urban traffic. IEEE Trans. Intell. Transp. Syst. **12**(3), 920–939 (2011)
9. American Association of State Highway and Transportation Officials. A Policy on Geometric Design of Highways and Streets (1994). American Association of State Highway and Transportation Officials, California (1995)
10. Paniati, J.F., Council, F.M.: The highway safety information system: transforming data into knowledge. Public Roads, **64**(3), 20–27 (2000). Winter
11. Elena, L., Mimbela, Y., Klein, L.A.: Summary of vehicle detection and surveillance technologies used in intelligent transportation systems, 31 August 2007
12. Senthil Kumar, T., Gautam, K.S., Haritha, H.: Debris detection and tracking system in water bodies using motion estimation technique. In: Snášel, V., Abraham, A., Krömer, P., Pant, M., Muda, A.K. (eds.) Innovations in Bio-Inspired Computing and Applications. AISC, vol. 424, pp. 275–284. Springer, Cham (2016). doi:10.1007/978-3-319-28031-8_24
13. https://en.wikipedia.org/wiki/induction-loop
14. Castillo Aguilar, J.J., Cabrera Carrillo, J.A., Guerra Fernández, A.J., Carabias Acosta, E.: Robust road condition detection system using in-vehicle standard sensors. Sensors **15**(12), 32056–32078 (2015). MDPI Journal
15. Tarabanis, K., Tsai, R.Y.: Computing occlusion-free viewpoints. In: Proceedings 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 802–807 (1992). doi:10.1109/CVPR.1992.223168
16. Betke, M., Haritaoglu, E., Davis, L.S.: Multiple vehicle detection and tracking in hard real-time. In: Proceedings of Conference on Intelligent Vehicles, pp. 351–356. IEEE Conference Publications (1996). doi:10.1109/IVS.1996.566405
17. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-based object tracking. IEEE Trans. Pattern Anal. Mach. Intell. **25**(5), 564–577 (2003). doi:10.1109/TPAMI.2003.1195991
18. Avidan, S., Shamir, A.: Seam carving for content-aware image resizing. ACM Trans. Graph. (TOG) **26**(3), 10 (2007). Proceedings of ACM SIGGRAPH 2007
19. Houben, Q., Carlos, J., Diaz, T., Debeir, O., Czyz, J.: Multi-feature stereo vision system for road traffic analysis. In: Proceedings of the Fourth International Conference on Computer Vision Theory and Applications, VISAPP 2009, Lisboa, Portugal, vol. 2, 5–8 February 2009
20. Hsieh, J.-W., Yu, S.-H., Chen, Y.-S., Hu, W.-F.: Automatic traffic surveillance system for vehicle tracking and classification. IEEE Trans. Intell. Transp. Syst. **7**(2), 175–187 (2006)

The Influence of Archive Size to SHADE

Adam Viktorin(✉), Roman Senkerik, Michal Pluhacek,
and Tomas Kadavy

Faculty of Applied Informatics, Tomas Bata University in Zlin,
T. G. Masaryka 5555, 760 01 Zlin, Czech Republic
{aviktorin, senkerik, pluhacek, kadavy}@fai.utb.cz

Abstract. This research analyzes the current archive of inferior solutions used in Success-History based Adaptive Differential Evolution (SHADE) and tests its influence on the result of optimization. A novel Enhanced Archive (EA) is analyzed in the same way and the results are compared. In order to compare both methods on different types of test functions, CEC2015 benchmark set was used as the test bed. Results suggest that there is a possibility for further research because current existing archive implementations do not provide sufficient benefits to the optimization algorithm.

Keywords: Differential Evolution · SHADE · Archive · Size

1 Introduction

Since 1995 [1], Differential Evolution (DE) has been a powerful tool for numerical optimization and it has been researched by a vast number of researchers all over the world. The main goal of such research is to produce an algorithm with better performance in terms of ability to find the global optima for an arbitrary problem. Over the years, an enormous number of DE variants emerged and thus, survey studies were performed [2–4].

One of the most successful directions in the DE research is an attempt to overcome the No Free Lunch (NFL) theorem [5] by adapting the control parameters – population size *NP*, scaling factor *F* and crossover rate *CR*. In the canonical version of DE [1], these parameters are set by the user along with the stopping criteria, which leaves too much space for an error in parameter settings that might lead to the bad performance of selected algorithm. The fine-tuning of control parameters is a time-consuming task and therefore, researchers have come up with various adaptive strategies. Examples of algorithms with adaptation of control parameters are: SDE [6], jDE [7], MDE_pBX [8], SaDE [9], JADE [10] and SHADE [11].

In JADE, crossover rate *CR* and scaling factor *F* are generated from Gaussian and Cauchy distributions respectively and mean values for these distributions are based on successful values of *CR* and *F* from the previous generation. Authors of JADE also proposed a novel mutation strategy "current-to-*p*best/1", which uses optional external archive of inferior solutions *A*. This archive serves to improve the diversity of the population, which should compensate the premature convergence, but no further

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_51

analysis of the archive influence was provided and the recommended archive size was the same as population size.

JADE formed a base for one of the most successful variants of adaptive DE – Success-History based Adaptive Differential Evolution (SHADE). On CEC2013 competition on real parameter single-objective optimization [12], SHADE placed 3rd and was the best performing DE based algorithm. SHADE variant with a linear decrease in population size titled L-SHADE [13] placed 1st next year on CEC2014 competition [14]. SHADE proposed additional memories to store historically successful *CR* and *F* values but uses still the same mutation strategy with an external archive. Thus, it is used in this paper to experimentally test the influence of archive size to the overall performance of the algorithm on CEC2015 benchmark set [15] and a novel Enhanced Archive (EA) implementation is tested as well.

The remainder of this paper is structured as follows: Sect. 2 covers DE, Sect. 3 describes SHADE algorithm and EA, experimental settings are depicted in Sect. 4, results are presented discussed in Sect. 5 and the whole paper is concluded in Sect. 6.

2 Differential Evolution

The DE algorithm is initialized with a random population of individuals $\boldsymbol{P}$, that represent solutions of the optimization problem. The population size *NP* is set by the user along with other control parameters – scaling factor *F* and crossover rate *CR*. In continuous optimization, each individual is composed of a vector $\boldsymbol{x}$ of length *D*, which is a dimensionality (number of optimized attributes) of the problem, and each vector component represents a value of the corresponding attribute, and of objective function value $f(\boldsymbol{x})$. For each individual in a population, three mutually different individuals are selected for mutation of vectors and resulting mutated vector v is combined with the original vector $\boldsymbol{x}$ in crossover step. The objective function value $f(\boldsymbol{u})$ of the resulting trial vector $\boldsymbol{u}$ is evaluated and compared to that of the original individual. When the quality (objective function value) of the trial individual is better, it is placed into the next generation, otherwise, the original individual is placed there. This step is called selection. The process is repeated until the stopping criterion is met (e.g. the maximum number of objective function evaluations, the maximum number of generations, the low bound for diversity between objective function values in population). The following sections describe four steps of DE: Initialization, mutation, crossover and selection.

2.1 Initialization

As aforementioned, the initial population $\boldsymbol{P}$, of size *NP*, of individuals is randomly generated. For this purpose, the individual vector $\boldsymbol{x}_i$ components are generated by Random Number Generator (RNG) with uniform distribution from the range which is specified for the problem by *lower* and *upper* bound (1).

$$\boldsymbol{x}_{j,i} = U\left[lower_j, upper_j\right] \text{ for } j = 1, \ldots, D \tag{1}$$

where i is the index of a current individual, j is the index of current attribute and D is the dimensionality of the problem.

In the initialization phase, a scaling factor value F and crossover value CR has to be assigned as well. The typical range for F value is [0, 2] and for CR, it is [0, 1].

2.2 Mutation

In the mutation step, three mutually different individuals $\boldsymbol{x}_{r1}$, $\boldsymbol{x}_{r2}$, $\boldsymbol{x}_{r3}$ from a population are randomly selected and combined in mutation according to the mutation strategy. The original mutation strategy of canonical DE is "rand/1" and is depicted in (2).

$$\boldsymbol{v}_i = \boldsymbol{x}_{r1} + F(\boldsymbol{x}_{r2} - \boldsymbol{x}_{r3}) \tag{2}$$

where $r1 \neq r2 \neq r3 \neq i$, F is the scaling factor value and $\boldsymbol{v}_i$ is the resulting mutated vector.

2.3 Crossover

In the crossover step, mutated vector $\boldsymbol{v}_i$ is combined with the original vector $\boldsymbol{x}_i$ and produces trial vector $\boldsymbol{u}_i$. The binary crossover (3) is used in canonical DE.

$$\boldsymbol{u}_{j,i} = \begin{cases} \boldsymbol{v}_{j,i} & \text{if } \; U[0,1] \leq CR \text{ or } j = j_{rand} \\ \boldsymbol{x}_{j,i} & \text{otherwise} \end{cases} \tag{3}$$

where CR is the used crossover rate value and j_{rand} is an index of an attribute that has to be from the mutated vector $\boldsymbol{u}_i$ (ensures generation of a vector with at least one new component).

2.4 Selection

The selection step ensures, that the optimization progress will lead to better solutions because it allows only individuals of better or at least equal objective function value to proceed into next generation $G + 1$ (4).

$$\boldsymbol{x}_{i,G+1} = \begin{cases} \boldsymbol{u}_{i,G} & \text{if } f\left(\boldsymbol{u}_{i,G}\right) \leq f\left(\boldsymbol{x}_{i,G}\right) \\ \boldsymbol{x}_{i,G} & \text{otherwise} \end{cases} \tag{4}$$

where G is the index of current generation.

The whole DE algorithm is depicted in pseudo-code below.

Algorithm pseudo-code 1: DE

```
Set $NP$, $CR$, $F$ and stopping criterion;
$G = 0$, $x_{best} = \{\}$;
Randomly initialize (1) population $\boldsymbol{P} = (\boldsymbol{x}_{1,G},\ldots,\boldsymbol{x}_{NP,G})$;
$\boldsymbol{P}_{new} = \{\}$, $\boldsymbol{x}_{best}$ = best from population $\boldsymbol{P}$;
while stopping criterion not met
  for $i = 1$ to $NP$ do
    $\boldsymbol{x}_{i,G} = \boldsymbol{P}[i]$;
    $\boldsymbol{v}_{i,G}$ by mutation (2);
    $\boldsymbol{u}_{i,G}$ by crossover (3);
    if $f(\boldsymbol{u}_{i,G}) < f(\boldsymbol{x}_{i,G})$ then
      $\boldsymbol{x}_{i,G+1} = \boldsymbol{u}_{i,G}$;
    else
      $\boldsymbol{x}_{i,G+1} = \boldsymbol{x}_{i,G}$;
    end
    $\boldsymbol{x}_{i,G+1} \rightarrow \boldsymbol{P}_{new}$;
  end
  $\boldsymbol{P} = \boldsymbol{P}_{new}$, $\boldsymbol{P}_{new} = \{\}$, $\boldsymbol{x}_{best}$ = best from population $\boldsymbol{P}$;
end
return $\boldsymbol{x}_{best}$ as the best found solution
```

3 Success-History Based Adaptive Differential Evolution and Enhanced Archive

In SHADE, the only control parameter that can be set by the user is population size *NP*, other two (*F, CR*) are adapted to the given optimization task, a new parameter *H* is introduced, which determines the size of *F* and *CR* value memories. The initialization step of the SHADE is, therefore, similar to DE. Mutation, however, is completely different because of the used strategy "current-to-*p*best/1" and the fact, that it uses different scaling factor value F_i for each individual. Crossover is still binary, but similarly to the mutation and scaling factor values, crossover rate value CR_i is also different for each individual. The selection step is the same and therefore following sections describe only different aspects of initialization, mutation and crossover steps. The last section is devoted to the proposed EA and its built into the SHADE algorithm.

3.1 Initialization

As aforementioned, initial population $\boldsymbol{P}$ is randomly generated as in DE, but additional memories for F and CR values are initialized as well. Both memories have the same size H and are equally initialized, the memory for CR values is titled $\boldsymbol{M}_{CR}$ and the memory for F is titled $\boldsymbol{M}_{F}$. Their initialization is depicted in (5).

$$\boldsymbol{M}_{CR,i} = \boldsymbol{M}_{F,i} = 0.5 \text{ for } i = 1, \ldots, H \tag{5}$$

Also, the external archive of inferior solutions $\boldsymbol{A}$ is initialized. Since there are no solutions so far, it is initialized empty $\boldsymbol{A}$ = Ø and its maximum size is set to NP.

3.2 Mutation

Mutation strategy "current-to-*p*best/1" was introduced in [9] and unlike "rand/1", it combines four mutually different vectors, therefore $pbest \neq r1 \neq r2 \neq i$ (6).

$$\boldsymbol{v}_i = \boldsymbol{x}_i + F_i\left(\boldsymbol{x}_{pbest} - \boldsymbol{x}_i\right) + F_i(\boldsymbol{x}_{r1} - \boldsymbol{x}_{r2}) \tag{6}$$

where $\boldsymbol{x}_{pbest}$ is randomly selected from the best $NP \times p$ best individuals in the current population. The p value is randomly generated for each mutation by RNG with uniform distribution from the range [p_{min}, 0.2]. where p_{min} = 2/NP. Vector $\boldsymbol{x}_{r1}$ is randomly selected from the current population and vector $\boldsymbol{x}_{r2}$ is randomly selected from the union of current population $\boldsymbol{P}$ and archive $\boldsymbol{A}$. The scaling factor value F_i is given by (7).

$$F_i = C\left[\boldsymbol{M}_{F,r}, 0.1\right] \tag{7}$$

where $\boldsymbol{M}_{F,r}$ is a randomly selected value (by index r) from $\boldsymbol{M}_F$ memory and C stands for Cauchy distribution, therefore the F_i value is generated from the Cauchy distribution with location parameter value $\boldsymbol{M}_{F,r}$ and scale parameter value 0.1. If the generated value $F_i > 1$, it is truncated to 1 and if it is $F_i \leq 0$, it is generated again by (7).

3.3 Crossover

Crossover is the same as in (3), but the CR value is changed to CR_i, which is generated separately for each individual (8). The value is generated from the Gaussian distribution with mean parameter value of $\boldsymbol{M}_{CR.r}$, which is randomly selected (by the same index r as in mutation) from $\boldsymbol{M}_{CR}$ memory and standard deviation value of 0.1.

$$CR_i = N\left[\boldsymbol{M}_{CR,r}, 0.1\right] \tag{8}$$

3.4 Historical Memory Updates

Historical memories $\boldsymbol{M}_F$ and $\boldsymbol{M}_{CR}$ are initialized according to (5), but its components change during the evolution. These memories serve to hold successful values of F and CR used in mutation and crossover steps. Successful in terms of producing trial individual better than the original individual. During one generation, these successful values are stored in corresponding arrays $\boldsymbol{S}_F$ and $\boldsymbol{S}_{CR}$. After each generation, one cell of $\boldsymbol{M}_F$ and $\boldsymbol{M}_{CR}$ memories is updated. This cell is given by the index k, which starts at 1 and increases by 1 after each generation. When it overflows the size limit of memories H, it is again set to 1. The new value of k-th cell for $\boldsymbol{M}_F$ is calculated by (9) and for $\boldsymbol{M}_{CR}$ by (10).

$$\boldsymbol{M}_{F,k} = \begin{cases} \text{mean}_{WL}(\boldsymbol{S}_F) & \text{if } \boldsymbol{S}_F \neq \emptyset \\ \boldsymbol{M}_{F,k} & \text{otherwise} \end{cases} \tag{9}$$

$$\boldsymbol{M}_{CR,k} = \begin{cases} \text{mean}_{WA}(\boldsymbol{S}_{CR}) & \text{if } \boldsymbol{S}_{CR} \neq \emptyset \\ \boldsymbol{M}_{CR,k} & \text{otherwise} \end{cases} \tag{10}$$

where $\text{mean}_{WL}()$ and $\text{mean}_{WA}()$ are weighted Lehmer (11) and weighted arithmetic (12) means correspondingly.

$$\text{mean}_{WL}(\boldsymbol{S}_F) = \frac{\sum_{k=1}^{|\boldsymbol{S}_F|} \boldsymbol{w}_k \cdot \boldsymbol{S}_{F,k}^2}{\sum_{k=1}^{|\boldsymbol{S}_F|} \boldsymbol{w}_k \cdot \boldsymbol{S}_{F,k}} \tag{11}$$

$$\text{mean}_{WA}(\boldsymbol{S}_{CR}) = \sum_{k=1}^{|\boldsymbol{S}_{CR}|} \boldsymbol{w}_k \cdot \boldsymbol{S}_{CR,k} \tag{12}$$

where the weight vector $\boldsymbol{w}$ is given by (13) and is based on the improvement in objective function value between trial and original individuals.

$$\boldsymbol{w}_k = \frac{\text{abs}\left(f\left(\boldsymbol{u}_{k,G}\right) - f\left(\boldsymbol{x}_{k,G}\right)\right)}{\sum_{m=1}^{|\boldsymbol{S}_{CR}|} \text{abs}\left(f\left(\boldsymbol{u}_{m,G}\right) - f\left(\boldsymbol{x}_{m,G}\right)\right)} \tag{13}$$

And since both arrays $\boldsymbol{S}_F$ and $\boldsymbol{S}_{CR}$ have the same size, it is arbitrary which size will be used for the upper boundary for m in (13). Complete SHADE algorithm is depicted in pseudo-code below.

Algorithm pseudo-code 2: SHADE

```
Set $NP$, $H$ and stopping criterion;
$G = 0$, $x_{best} = \{\}$, $k = 1$, $p_{min} = 2/NP$, $\mathbf{A} = \emptyset$;
Randomly initialize (1) population $\mathbf{P} = (\mathbf{x}_{1,G},\ldots,\mathbf{x}_{NP,G})$;
Set $\mathbf{M}_F$ and $\mathbf{M}_{CR}$ according to (5);
$\mathbf{P}_{new} = \{\}$, $\mathbf{x}_{best}$ = best from population $\mathbf{P}$;
while stopping criterion not met
  $\mathbf{S}_F = \emptyset$, $\mathbf{S}_{CR} = \emptyset$;
  for $i$ = 1 to $NP$ do
    $\mathbf{x}_{i,G} = \mathbf{P}[i]$;
    $r = U[1, H]$, $p_i = U[p_{min}, 0.2]$;
    Set $F_i$ by (7) and $CR_i$ by (8);
    $\mathbf{v}_{i,G}$ by mutation (6);
    $\mathbf{u}_{i,G}$ by crossover (3);
    if $f(\mathbf{u}_{i,G}) < f(\mathbf{x}_{i,G})$ then
      $\mathbf{x}_{i,G+1} = \mathbf{u}_{i,G}$;
      $\mathbf{x}_{i,G} \rightarrow \mathbf{A}$;
      $F_i \rightarrow \mathbf{S}_F$, $CR_i \rightarrow \mathbf{S}_{CR}$;
    else
      $\mathbf{x}_{i,G+1} = \mathbf{x}_{i,G}$;
    end
    if $|\mathbf{A}| > NP$ then randomly delete an ind. from $\mathbf{A}$;
    $\mathbf{x}_{i,G+1} \rightarrow \mathbf{P}_{new}$;
  end
  if $\mathbf{S}_F \neq \emptyset$ and $\mathbf{S}_{CR} \neq \emptyset$ then
    Update $\mathbf{M}_{F,k}$ (9) and $\mathbf{M}_{CR,k}$ (10), $k$++;
    if $k > H$ then $k = 1$, end;
  end
  $\mathbf{P} = \mathbf{P}_{new}$, $\mathbf{P}_{new} = \{\}$, $\mathbf{x}_{best}$ = best from population $\mathbf{P}$;
end
return $\mathbf{x}_{best}$ as the best found solution
```

3.5 Enhanced Archive

The original archive of inferior solutions $\boldsymbol{A}$ in SHADE is filled during the selection step and contains original individuals, that had worse objective function value than trial individuals produced from them. The maximum size of the archive is NP and wherever it overflows, a random individual is removed from the archive.

During the preliminary testing, results shown, that the frequency of successful uses of individuals from the archive is quite low and that there might be a possibility of implementing more beneficial archive solution. Therefore, the EA was proposed as an alternative which was inspired in the field of discrete optimization. The basic idea is, that trial individuals unsuccessful in selection should not be discarded as bad solutions, but the best of them should be stored in the archive in order to provide promising search directions. This is done by placing only unsuccessful trial individuals in the archive and when the archive overflows the maximum size, the worst individual (in terms of objective function value) is discarded. Thus, the original SHADE algorithm was partially changed to implement this novel archive and the changes in pseudo-code are depicted below.

```
Algorithm pseudo-code 3: SHADE changes to accommodate EA
    if $f(\mathbf{u}_{i,G}) < f(\mathbf{x}_{i,G})$ then
      $\mathbf{x}_{i,G+1} = \mathbf{u}_{i,G}$;
      ~~$\mathbf{x}_{i,G} \rightarrow \mathbf{A}$~~;
      $F_i \rightarrow \mathbf{S}_F$, $CR_i \rightarrow \mathbf{S}_{CR}$;
    else
      $\mathbf{x}_{i,G+1} = \mathbf{x}_{i,G}$, $\mathbf{u}_{i,G} \rightarrow \mathbf{A}$;
    end
    if $|\mathbf{A}|>NP$ then delete the worst individual from $\mathbf{A}$;
    $\mathbf{x}_{i,G+1} \rightarrow \mathbf{P}_{new}$;
```

4 Experimental Setting

Since the archive in SHADE algorithm can be understand as the second population, its size might be an important factor in the optimization. The archive is used for one individual present in mutation and when the archive size is the same as the population size, the probability of archive use is approximately 50% (depends on the implementation). But with varying archive size, the probability changes. The probability is connected to the ratio between population size NP and archive size $|A|$ and is equal to $|A|/(NP + |A|)$.

In this work, Original Archive (OA) and EA sizes ranged from 0 to 100 with the step of 10 and the population size NP was 100. The dimension of CEC2015 benchmark functions was set to 10 and 51 independent runs were performed with the stopping criteria of 100,000 test function evaluations. SHADE historical memory size H was set to 10.

5 Results and Discussion

The performance of SHADE algorithm with different archive implementation and sizes was compared on CEC2015 benchmark set, where the mean error values of 51 runs were used. Friedman test for different sizes of OA yielded the p-value of 0.44 and similar value 0.43 was yielded for EA, therefore the hypothesis, that there are no significant differences in the performance with different size of the archive cannot be rejected on the significance level of 0.05. Individual comparison of the algorithm with different archive size was done by Wilcoxon signed rank test, with a significance level

Table 1. Wilcoxon signed rank test (α = 0.05) results for different sizes of original archive in SHADE algorithm on CEC2015 benchmark set.

|A|	0	10	20	30	40	50	60	70	80	90	100
0	=	=	=	=	=	=	=	=	=	=	=
10	=	=	<	<	=	=	=	<	=	<	<
20	=	>	=	=	=	=	=	=	=	=	=
30	=	>	=	=	=	=	>	=	=	=	=
40	=	=	=	=	=	=	=	=	=	=	=
50	=	=	=	=	=	=	=	=	=	=	=
60	=	=	=	<	=	=	=	<	=	<	<
70	=	>	=	=	=	=	>	=	=	=	=
80	=	=	=	=	=	=	=	=	=	=	=
90	=	>	=	=	=	=	>	=	=	=	=
100	=	>	=	=	=	=	>	=	=	=	=
Score	0	5	−1	−2	0	0	4	−2	0	−2	−2

Table 2. Wilcoxon signed rank test (α = 0.05) results for different sizes of enhanced archive in SHADE algorithm on CEC2015 benchmark set.

|A|	0	10	20	30	40	50	60	70	80	90	100
0	=	=	=	=	=	=	=	=	=	=	=
10	=	=	=	=	=	=	=	>	=	=	=
20	=	=	=	=	=	<	=	=	=	=	=
30	=	=	=	=	=	=	<	=	=	=	=
40	=	=	=	=	=	>	=	>	=	=	>
50	=	=	>	=	<	=	=	=	=	=	=
60	=	=	=	>	=	=	=	=	=	=	>
70	=	<	=	=	<	=	=	=	=	<	=
80	=	=	=	=	=	=	=	=	=	=	=
90	=	=	=	=	=	=	=	>	=	=	=
100	=	=	=	=	<	=	<	=	=	=	=
Score	0	−1	1	1	−3	0	−1	3	0	−1	2

of 0.05 and the results are presented in Table 1 for OA, and Table 2 for EA. In these tables, symbol = stands for no significant difference, symbol <stands for a significantly better result of archive size in the row and symbol> for a significantly better result of archive size in the column. The last row in tables depicts the sum of +better and –worse results.

In Table 1, the most promising archive sizes are 10 and 60, where they score 5 and 4 respectively, meaning, that these sizes are beneficial in 5 and 4 out of 10 cases. On the other hand, there is no single worst performing archive size, but rather a group with score of −2. In Table 2, the differences are even smaller. Most promising archive size is 70 with the score of 3 and the worst archive size is 40 with the score of −3.

The most interesting result is shown in the second column of both tables. There is no significant difference in the results when the archive is not used at all – its size is 0.

When the results were compared altogether, the best settings were still original archives with sizes 10 and 60, with scores of 8 and 9 out of 21 and the worst were an original archive with size 90 and an enhanced archive with size 60, with the score of −5. Nevertheless, there was still no significant difference when there was no archive at all. Therefore, it seems that the diversity of the population or the ability to avoid the premature convergence are not superior with the current external archive implementations to the algorithm with no archive.

6 Conclusion

This paper presented an archive size analysis in SHADE algorithm and also an analysis of the novel archive implementation called enhanced archive. Both archive implementations were tested on CEC2015 benchmark set and results were statistically tested.

Results presented in this paper shown that there are some beneficial settings for the archive, but there is none that would perform significantly better than the SHADE algorithm without an archive. This suggests, that there is still a possibility in research of external archive implementation, that would actually help the optimization process in escaping the local optima and avoiding premature convergence. Thus, the future research will be aimed in that direction and also a similar analysis will be done in higher dimensional spaces.

Acknowledgements. This work was supported by Grant Agency of the Czech Republic – GACR P103/15/06700S, further by the Ministry of Education, Youth and Sports of the Czech Republic within the National Sustainability Programme Project no. LO1303 (MSMT-7778/2014). Also by the European Regional Development Fund under the Project CEBIA-Tech no. CZ.1.05/2.1.00/03.0089 and by Internal Grant Agency of Tomas Bata University under the Projects no. IGA/CebiaTech/2017/004.

References

1. Storn, R., Price, K.: Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces, vol. 3. ICSI, Berkeley (1995)
2. Neri, F., Tirronen, V.: Recent advances in differential evolution: a survey and experimental analysis. Artif. Intell. Rev. **33**(1–2), 61–106 (2010)
3. Das, S., Suganthan, P.N.: Differential evolution: a survey of the state-of-the-art. IEEE Trans. Evol. Comput. **15**(1), 4–31 (2011)
4. Das, S., Mullick, S.S., Suganthan, P.N.: Recent advances in differential evolution–an updated survey. Swarm Evol. Comput. **27**, 1–30 (2016)
5. Wolpert, D.H., Macready, W.G.: No free lunch theorems for optimization. IEEE Trans. Evol. Comput. **1**(1), 67–82 (1997)
6. Omran, Mahamed G.H., Salman, A., Engelbrecht, Andries P.: Self-adaptive differential evolution. In: Hao, Y., Liu, J., Wang, Y., Cheung, Y.-M., Yin, H., Jiao, L., Ma, J., Jiao, Y.-C. (eds.) CIS 2005. LNCS (LNAI), vol. 3801, pp. 192–199. Springer, Heidelberg (2005). doi:10.1007/11596448_28

7. Brest, J., Greiner, S., Bošković, B., Mernik, M., Zumer, V.: Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. IEEE Trans. Evol. Comput. **10**(6), 646–657 (2006)
8. Islam, S.M., Das, S., Ghosh, S., Roy, S., Suganthan, P.N.: An adaptive differential evolution algorithm with novel mutation and crossover strategies for global numerical optimization. IEEE Trans. Syst. Man Cybern. Part B (Cybern.) **42**(2), 482–500 (2012)
9. 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)
10. Zhang, J., Sanderson, A.C.: JADE: adaptive differential evolution with optional external archive. IEEE Trans. Evol. Comput. **13**(5), 945–958 (2009)
11. Tanabe, R., Fukunaga, A.: Success-history based parameter adaptation for differential evolution. In: 2013 IEEE Congress on Evolutionary Computation (CEC), pp. 71–78. IEEE, June 2013
12. Liang, J.J., Qu, B.Y., Suganthan, P.N., Hernández-Díaz, A.G.: Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization. Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Nanyang Technological University, Singapore, Technical report, 201212 (2013)
13. Tanabe, R., Fukunaga, A.S.: Improving the search performance of SHADE using linear population size reduction. In: 2014 IEEE Congress on Evolutionary Computation (CEC), pp. 1658–1665. IEEE, July 2014
14. Liang, J.J., Qu, B.Y., Suganthan, P.N.: Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization. Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore (2013)
15. Liang, J.J., Qu, B.Y., Suganthan, P.N., Chen, Q.: Problem definitions and evaluation criteria for the CEC 2015 competition on learning-based real-parameter single objective optimization. Technical Report201411A, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical report, Nanyang Technological University, Singapore (2014)

Comparing Border Strategies for Roaming Particles on Single and Multi-swarm PSO

Tomas Kadavy(✉), Michal Pluhacek, Adam Viktorin,
and Roman Senkerik

Faculty of Applied Informatics, Tomas Bata University in Zlin,
T.G. Masaryka 5555, 760 01 Zlin, Czech Republic
{kadavy, pluhacek, aviktorin, Senkerik}@fai.utb.cz

Abstract. In this paper, the methods for handling particles that violate available search spaces are compared using a single and multi-swarm technique. The methods are soft borders and hypersphere universe. The goal is to compare this approaches and its combination. The comparisons are made on CEC'17 benchmark set functions. The experiments were carried out according to CEC benchmark rules and statistically evaluated.

Keywords: Particle swarm optimization · PSO · Search space boundaries · Multi-swarm · Roaming particles

1 Introduction

Despite the fact that Particle Swarm Optimization (PSO) was first proposed in 1995 [1], its well-known weaknesses are still actual and many researches are working on improving that. Classical PSO have numerous adjustment options. These can be: learning factors, inertia weight, maximum velocity and others. In this paper, the strategies for handling roaming particles are compared with their impact on solutions quality. The boundaries of the search space are defined by particular optimization problem and define the minimally acceptable and maximal acceptable value of the solutions in dimensions. The compared strategies in this paper are: soft borders and hypersphere universe. For the hyperspace method, the velocity formula can by changed to match a unique characteristic of this method. This velocity update is categorized as a different strategy among to hypersphere universe using only classical velocity formula. The strategies in this paper are compared on CEC'17 benchmark set [2]. The strategies are also combined together using multi-swarm technique [3].

The paper is structured as follows. The PSO and its multi-swarm modification are described in Sect. 2. The methods for handling the roaming particles are described in Sect. 3. The experiment setup is detailed in Sect. 4. Section 5 contains statistical overviews of results and performance comparisons obtained during the evaluation on benchmark set. Discussion and conclusion follow.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_52

2 Particle Swarm Optimization

A typical representative of swarm intelligence algorithms is Particle Swarm Optimization (PSO). This algorithm was published in 1995 by Ebenhart and Kennedy in [1]. This algorithm mimics the social behavior and movement of swarm members. Originally it was bird flocking and fish schooling. Because it is a quite long time from his first appearance, the numerous versions appeared. These new versions mostly want to improve the fact that PSO has well know weakness, the premature convergence to local optima.

In this PSO algorithm, the individuals (also called particles) are moving in space of possible solutions of the defined particular problem. This movement is determined by two factors. One of them is the current position of a particle, labelled as x. The second one is the velocity of a particle, labelled as v. Each particle also remembers his best position (solution of the problem) obtained so far. This solution is tagged as the *pBest*, personal best solution. Also, each particle has access to the global best solution, *gBest*, which is selected from all *pBests*. These variables set the direction for every particle and then even a new position in next iteration. PSO usually stops after a number of iterations, or a number of FEs, fitness evaluations, defined by the user.

The particle position is represented as coordinates in n-dimensional space of solutions. These coordinates are parameters of the optimized problem. In every step of the algorithm, the new positions of particles are calculated based on previous positions and velocities. The new position of a particle is checked if it still lies in space of possible solutions. The function of the optimized problem is called Cost Function (CF). The position of a particle is used as input parameters in CF. If the value of CF is better than the value saved in *pBest* of a particle, then the particle saves this new position as his new *pBest*. Also, if this *pBest* has better CF value than the previous *gBest*, then this *pBest* is saved as new *gBest*.

The position of particle x is calculated according the formula (1)

$$x_{ij}^{t+1} = x_{ij}^{t} + v_{ij}^{t+1} \tag{1}$$

where $t + 1$ is actual iteration, t is then previous iteration, x_{ij} is the position of a i-particle in j-dimension, v is the velocity of a particle.

The velocity of a particle v is calculated according to (2).

$$v_{ij}^{t+1} = w \cdot v_{ij}^{t} + c_1 \cdot r_1 \cdot \left(pBest_{ij} - x_{ij}^{t}\right) + c_2 \cdot r_2 \cdot \left(gBest_{j} - x_{ij}^{t}\right) \tag{2}$$

where w is inertia weight [4], c_1 and c_2 are learning factors and r_1 and r_2 are random numbers of unimodal distribution in range <0,1>.

2.1 Multi-swarm

Due to premature convergence of classical PSO, some techniques were developed. One of them is called multi-swarm [3]. The basic idea of this mechanism is splitting the swarm population into, typically two, subpopulations. Each group has own *gBest*. These subgroups exchange their *gBest* (*gBest* of better fitness value is copied to others subpopulations) every n-iterations, defined by the user. Each subpopulation can use different behaviors.

3 Strategies

When the position of a particle is updated, the particle has to be checked if its new position is in the appropriate boundaries (inside a space of possible solutions). If the particle is not in the appropriate boundaries (roaming particle), some corrections have to be made. There are several different strategies for this purpose. For this paper two of them are selected: soft borders and hypersphere universe.

3.1 Soft Borders

The particle can travel outside of the search space, there is only one restriction applied on that particle. His cost function is not calculated and therefore his *pBest* is unchanged. If a certain condition is met [5], the particle will eventually return inside a search space by itself.

3.2 Hypersphere Universe

This method simulates an endless spherical universe. For example, if a particle violates upper boundary of the search space, the particle than appear in the search space from lower boundary. The upper boundary is neighbouring the lower one of corresponding dimension and vice versa. This approach is explained in Fig. 1.

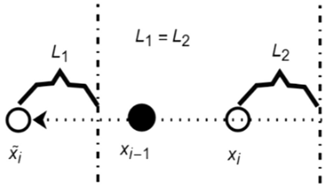


Fig. 1. Explanation of hypersphere universe method. x_{i-1} is the particle position in last iteration, $\overline{x_i}$ is uncorrected position and x_i is the final correct position.

Using this method, a typical way to compute velocity (2) of a particle becomes less efficient, because a particle can choose a longer way (vector of particle position and his *pBest* or *gBest*). The hypersphere universe offers the second option, a particle can travel through a boundary and reach the final destination using shorter vector. In Fig. 2 the *L1* is vector computed using the standard velocity update (2) and vector *L2* is a new vector that appears when the hyperspace method is used. The sum of these vectors is the range of available search space.

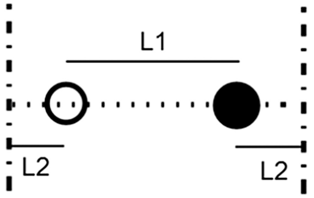


Fig. 2. Two possible velocity vectors in hypersphere universe.

The new velocity formula for this method, which can choose the better (smaller) vector, is defined as (3).

$$v_{ij}^{t+1} = w \cdot v_{ij}^{t} + c_1 \cdot r_1 \cdot L_{P,ij}^{t} + c_2 \cdot r_2 \cdot L_{G,ij}^{t} \tag{3}$$

where $L_{P,ij}^{t}$ and $L_{G,ij}^{t}$ are defined in formula (4).

$$\begin{gathered}
L_{P,ij} = \begin{cases} \widehat{L}_{P,ij}, & if\left|\widehat{L}_{P,ij}\right| \leq d \\ \widehat{L}_{P,ij}\, mod(-d), & if\left(\left|\widehat{L}_{P,ij}\right| > d \wedge \left|\widehat{L}_{P,ij}\right| > 0\right) \\ \widehat{L}_{P,ij}\, mod(+d), & if\left(\left|\widehat{L}_{P,ij}\right| > d \wedge \left|\widehat{L}_{P,ij}\right| \leq 0\right) \end{cases} \\
L_{G,ij} = \begin{cases} \widehat{L}_{G,ij}, & if\left|\widehat{L}_{G,ij}\right| \leq d \\ \widehat{L}_{G,ij}\, mod(-d), & if\left(\left|\widehat{L}_{G,ij}\right| > d \wedge \left|\widehat{L}_{G,ij}\right| > 0\right) \\ \widehat{L}_{G,ij}\, mod(+d), & if\left(\left|\widehat{L}_{G,ij}\right| > d \wedge \left|\widehat{L}_{G,ij}\right| \leq 0\right) \end{cases}
\end{gathered} \tag{4}$$

The d is computed by formula (5) where b^u and b^l stand for upper bound limit and lower bound limit of the search space. The $\widehat{L}_{P,ij}$ and $\widehat{L}_{G,ij}$ are defined in (6).

$$d = \frac{\left|b^u - b^l\right|}{2} \tag{5}$$

$$\begin{gathered} \widehat{L}_{P,ij} = pBest_{ij} - x_{ij} \\ \widehat{L}_{G,ij} = gBest_{j} - x_{ij} \end{gathered} \tag{6}$$

4 Experimental Setup

The experiments were performed for dimension dim = 10 on CEC'17 benchmark functions set [2]. The maximal number of cost function evaluations is set to $10000 \cdot dim$ according to the definition for this benchmark set. The population size (NP) is set to 40 for all dimensions. The inertia weight is set w = 0.729 and learning factors are

$c_1 = c_2 = 1.49445$ according to [5]. Every test function is repeated for 51 independent runs and the results are statistically evaluated. The benchmark set includes 30 functions separated into four categories: unimodal, multimodal, hybrid and composite. Each function has search space defined in $[-100{,}100]^{Dim}$ and global minimum is $100 \cdot f_i$, where i is an order of test function f. In Table 1 is shown the summary of tested functions.

Table 1. Tested functions

Test function f_i	Category	Global minimum
f_1	Unimodal	100
f_2		200
f_3		300
f_4	Multimodal	400
f_5		500
f_6		600
f_7		700
f_8		800
f_9		900
f_{10}		1000
f_{11}	Hybrid	1100
f_{12}		1200
f_{13}		1300
f_{14}		1400
f_{15}		1500
f_{16}		1600
f_{17}		1700
f_{18}		1800
f_{19}		1900
f_{20}		2000
f_{21}	Composition	2100
f_{22}		2200
f_{23}		2300
f_{24}		2400
f_{25}		2500
f_{26}		2600
f_{27}		2700
f_{28}		2800
f_{29}		2900
f_{30}		3000

The six variants were tested. Three of them are multi-swarms where each of them is composed of two subpopulations of 20 particles and every 100 iterations the subpopulations compare their *gBest* and both of them use the better one (*gBest* with

Table 2. Variant description

Variant	Type	Border strategy	
A1	Single-swarm	Soft border	
A2	Single-swarm	Hypersphere	
A3	Single-swarm	Hypersphere with velocity formula (3)	
A4	Multi-swarm	Soft border	Hypersphere
A5	Multi-swarm	Soft border	Hypersphere with velocity formula (3)
A6	Multi-swarm	Hypersphere	Hypersphere with velocity formula (3)

smaller fitness value). Each variant uses different border strategy for its subpopulations. In Table 2 is the detailed list of this setting.

5 Results

The Friedman test [6] was used for statistical comparison of used variants. To compute the statistics, the JAVA package from 'http://sci2s.ugr.es/keel/multipleTest.zip' was used. The results of each test function from CEC benchmark set were averaged from their 51 independent runs. These average results were used for the Friedman test.

The p-value computed by Friedman test is 4.13E−8, the critical value of the Friedman statistic is at $\alpha = 0.05$ [7] so the ranking of Friedman statistics is valid.

The non-parametric Friedman test ranking of the variants is in Table 3. The adjusted Holm p-values among A6 variant and others are shown in Table 4.

Table 3. Friedman ranking of variants

Variant	Ranking
A6	2.28
A5	2.45
A1	3.48
A3	3.62
A4	4.43
A2	4.73

Table 4. Adjusted p-values of A6 variant and others variants

Variant	p-value
A5	0.73E0
A1	0.03E0
A3	0.02E0
A4	3.42E−5
A2	1.97E−6

Furthermore, selected examples of mean *gBest* value history are shown in Figs. 3, 4, 5 and 6.

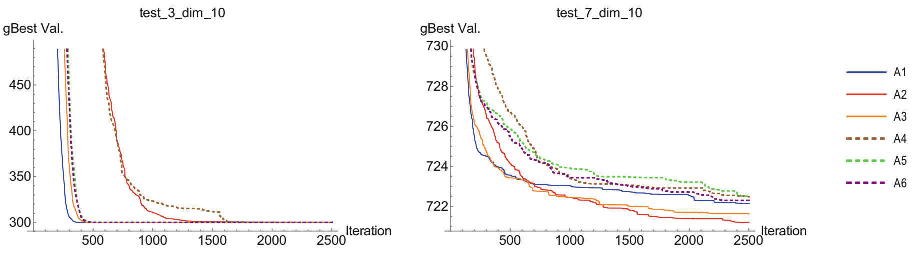


Fig. 3. Comparisons of gBests mean history over 51 runs

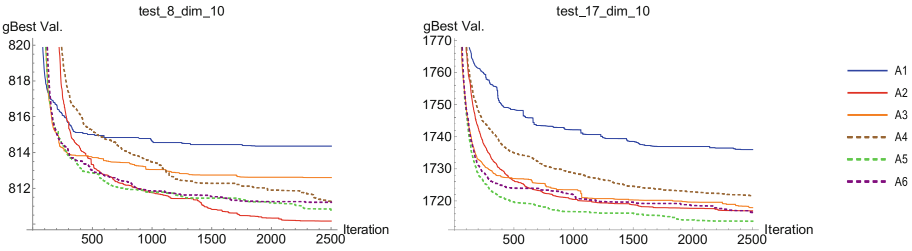


Fig. 4. Comparisons of gBests mean history over 51 runs

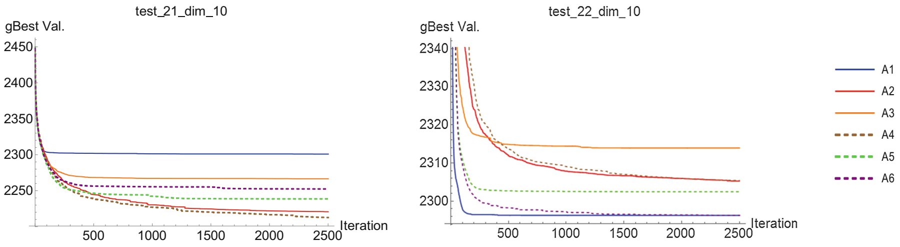


Fig. 5. Comparisons of gBests mean history over 51 runs

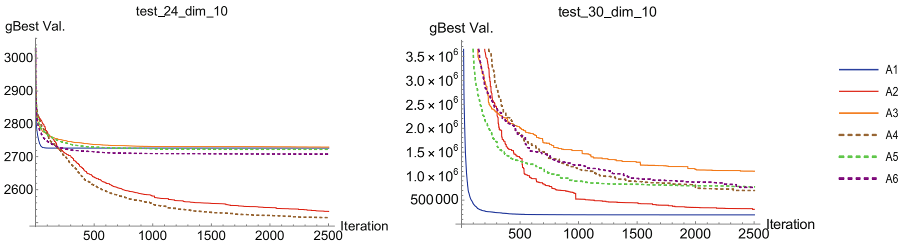


Fig. 6. Comparisons of gBests mean history over 51 runs

6 Results Discussion

The results in Table 3 are sorted in ascending order and the first variant is, therefore, the best performing one. According to Table 4, if the adjusted p-value is smaller than value 0.1 for Holm test, the compared version is significantly better in performance. With this values, the variants A1, A2, A3 and A4 are significantly different from the best performing variant A6. Between variants A6 and A5 there is no significant difference. With this knowledge, the best performing variants are A6 and A5 on tested benchmark functions.

This trend can be seen also in Figs. 4 and 5. In few cases, the *gBest* mean history show the opposite trend, but the difference with others variants are small. In the figures (Figs. 4 and 6), the variants A6 and A5 are close to each other on most test functions.

In Fig. 3, for test function 3 (unimodal), the convergence speed seems to be the slowest for the A2 and A4 variants, this trend can be caused due to the high velocity of particles.

7 Conclusion

In this paper, the results of six possible variants of handling particles position in n-dimensional solution space were presented. Three methods (soft borders, hypersphere and hypersphere with new velocity formula) were tested on the single swarm and their combination on multi-swarm techniques. The methods were tested on classical PSO algorithm. For comparison, the benchmark set CEC'17 was used. The results were presented and tested for statistical significance using Friedman test. Based on this results, some conclusions can be made.

The best performing variants were A6 and A5. Both variants are multi-swarm which have one common border strategy, the hypersphere universe with updated velocity equation.

The goal of this study was to show and compare differences in performance of the selected methods with their combinations using the multi-swarm technique. The results of this study will be further used in future studies to suggest possible improvements for controlling the position of particles that violates search space boundaries.

Acknowledgements. This work was supported by Grant Agency of the Czech Republic – GACR P103/15/06700S, further by the Ministry of Education, Youth and Sports of the Czech Republic within the National Sustainability Programme Project no. LO1303 (MSMT-7778/2014. Also by the European Regional Development Fund under the Project CEBIA-Tech no. CZ.1.05/2.1.00/03.0089 and by Internal Grant Agency of Tomas Bata University under the Projects no. IGA/CebiaTech/2017/004.

References

1. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Networks 1995, pp. 1942–1948 (1995)
2. Awad, N.H. et al.: Problem definitions and evaluation criteria for CEC 2017 special session and competition on single-objective real-parameter numerical optimization (2016)
3. Pluhacek, M., Senkerik, R., Viktorin, A., Zelinka, I.: Single swarm and simple multi-swarm PSO comparison. In: 2016 9th EUROSIM Congress on Modelling and Simulation, Oulu, pp. 498–502 (2016)
4. Kennedy, J.: The particle swarm: social adaptation of knowledge. In: Proceedings of the IEEE International Conference on Evolutionary Computation, pp. 303–308 (1997)
5. Eberhart, R.C., Shi, A.Y.: Comparing inertia weights and constriction factors in particle swarm optimization. In: Proceedings of the 2000 Congress on Evolutionary Computation, CEC 2000, pp. 84–88. IEEE (2000)
6. Friedman, M.: The use of ranks to avoid the assumption of normality implicit in the analysis of variance. J. Am. Stat. Assoc. **32**, 675–701 (1937)
7. Demsar, J.: Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. **7**, 1–30 (2006)

On the Randomization of Indices Selection for Differential Evolution

Roman Senkerik(✉), Michal Pluhacek, Adam Viktorin, and Tomas Kadavy

Faculty of Applied Informatics, Tomas Bata University in Zlin,
Nam T.G. Masaryka 5555, 760 01 Zlin, Czech Republic
{senkerik, pluhacek, aviktorin, kadavy}@fai.utb.cz

Abstract. This research deals with the hybridization of two softcomputing fields, which are the chaos theory and evolutionary algorithms. This paper investigates the utilization of the two-dimensional discrete chaotic systems, which are Burgers and Lozi maps, as the chaotic pseudo random number generators (CPRNGs) embedded into the selected heuristics, which is differential evolution algorithm (DE). Through the utilization of either chaotic systems or identical identified pseudo random number distribution, it is possible to fully keep or remove the hidden complex chaotic dynamics from the generated pseudo random data series. Experiments are focused on the extended investigation, whether the different randomization types with different pseudo random numbers distribution or hidden complex chaotic dynamics providing the unique sequencing are more beneficial to the heuristic performance. This research utilizes set of 4 selected benchmark functions, and totally four different randomizations; further results are compared against canonical DE.

Keywords: Differential evolution · Complex dynamics · Deterministic chaos · Randomization · Burgers map · Lozi map

1 Introduction

This research deals with the mutual intersection of the two softcomputing fields, which are the complex dynamics given by the chaotic systems driving the selection of indices in Differential Evolution (DE) algorithm and evolutionary computation techniques (ECT's). Currently the DE [1] is known as powerful heuristic for many difficult and complex optimization problems.

A number of DE variants have been recently developed with the emphasis on adaptivity/selfadaptivity [2], ensemble approach [3] or other modern approaches [4, 5]. The importance of randomization within heuristics as a compensation of limited amount of search moves is stated in the survey paper [6]. This idea has been carried out in subsequent studies describing different techniques to modify the randomization process [7, 8] and especially in [9], where the sampling of the points is tested from modified distribution. The importance and influence of randomization operations was also deeply experimentally tested in simple control parameter adjustment jDE strategy [10]. Together with this persistent development in such mainstream research topics, the

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_53

basic concept of chaos driven DE have been introduced. Recent research in chaotic approach for heuristics generally uses various chaotic maps in the place of pseudo random number generators (PRNG). The focus of this research is the direct embedding of chaotic dynamics in the form of chaos pseudo random number generator (CPRNG) for heuristic. The initial concept of embedding chaotic dynamics into the evolutionary/swarm algorithms is given in [11]. Later, the initial study [12] was focused on the simple embedding of chaotic systems for DE and Self Organizing Migration Algorithm (SOMA) [13]. Also the PSO (Particle Swarm Optimization) algorithm with elements of chaos was introduced as CPSO [14] followed by the introduction of chaos embedded PSO with inertia weigh strategy [15], further PSO strategy driven alternately by two chaotic systems [16] and finally PSO with ensemble of chaotic systems [17]. Recently the chaos driven heuristic concept has been utilized in ABC algorithm [18] and applications with DE [19].

The organization of this paper is following: Firstly, the motivation and novality for this research is proposed. The next sections are focused on the description of the concept of chaos driven DE, identification of chaotic series distribution and the experiment background. Results and conclusion follow afterwards.

2 Motivation

This research is an extension and continuation of the previous successful initial experiment with the single/multi-chaos driven DE (ChaosDE), where the positive influence of hidden complex dynamics for the heuristic performance has been experimentally shown. This research is also a follow up to previous initial experiments with time continuous chaotic systems and different sampling rates used [20].

Nevertheless, the questions remain, as to why it works, why it may be beneficial to use the correlated chaotic time series for generating pseudo random numbers driving the selection, mutation, crossover or other processes in particular heuristics.

The novality of the research is given by the experiment investigationg whether the chaos embedded heuristics concept belongs to the group of either "utilization of different PRNG with different distribution" or the unique chaos dynamics providing unique sequencing of pseudo random numbers is the key of performance improvements. The last point was also inspired by recent advances in connection of complexity and heuristic [21] together with the research focused on selection of indices in DE [22] where the indices (solutions) for mutation process were not selected randomly, but based on the complex behavior and neighborhood mechanisms.

To confirm or disprove the aforementioned hypothesis, a simple experiment was performed and presented here. Through the utilization of either chaotic systems or identical identified pseudo random number distribution, it is possible to fully keep or remove the hidden complex chaotic dynamics from the generated pseudo random data series for obtaining the pseudo random numbers for indices selection inside DE.

3 Differential Evolution

DE is a population-based optimization method that works on real-number-coded individuals [1]. DE is quite robust, fast, and effective, with global optimization ability. It does not require the objective function to be differentiable, and it works well even with noisy and time-dependent objective functions. There are essentially five inputs to the heuristic. *Dim* is the size of the problem, *Gmax* is the maximum number of generations, *NP* is the total number of solutions, *F* is the scaling factor of the solution and *CR* is the factor for crossover. *F* and *CR* together make the internal tuning parameters for the heuristic. Due to a limited space and the aims of this paper, the detailed description of well known canonical strategy of differential evolution algorithm basic principles is insignificant and hence omitted. Please refer to [1, 23] for the detailed description of the used *DE/Rand/1/Bin* strategy (both for ChaosDE and Canonical DE) as well as for the complete description of all other strategies.

4 Chaotic Systems and Identification of CPRNGs Distributions

Following two well known and frequently utilized discrete dissipative chaotic maps were used as the CPRNGS for DE: Burgers (1), and Lozi map (2).

The Burgers mapping is a discretization of a pair of coupled differential equations to illustrate the relevance of the concept of bifurcation to the study of hydrodynamics flows. The Lozi map is a simple discrete two-dimensional chaotic map. With the typical settings as in Table 1, systems exhibits typical chaotic behavior [24].

Table 1. Definition of chaotic systems used as CPRNGs

Chaotic maps equations	Parameter settings
$X_{n+1} = aX_n - Y_n^2$ $Y_{n+1} = bY_n + X_nY_n$ (1)	a = 0.75 and b = 1.75
$X_{n+1} = 1 - a\lvert X_n\rvert + bY_n$ $Y_{n+1} = X_n$ (2)	a = 1.7 and b = 0.5

For the comparisons of DE performance with indices selection driven either by CPRNG or identical PRNG distribution without chaotic dynamics, it was necessary to perform the CPRNGs distributions identification with 10000 samples and statistical distribution fit tests. *Statistica* and *Wolfram Mathematica* software were used for this task with following results (See also Figs. 1 and 2):

- Burgers map based CPRNG was identified as Beta distribution (α, β) with α = 0.63 and β = 3.54.
- Lozi map based CPRNG was identified as Beta distribution (α, β) with α = 1.05 and β = 1.57.

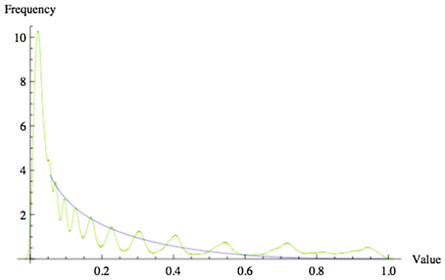


Fig. 1. Identification (blue line) of Burgers map based CPRNG (green line – smooth histogram)

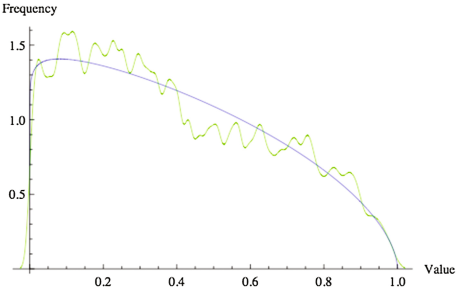


Fig. 2. Identification (blue line) of Lozi map based CPRNG (green line – smooth histogram)

5 The Concept of ChaosDE with Discrete Chaotic System as Driving CPRNG

The general idea of CPRNG is to replace the default PRNG with the chaotic system. As the chaotic system is a set of equations with a static start position, we created a random start position of the system, in order to have different start position for different experiments. Thus we are utilizing the typical feature of chaotic systems, which is extreme sensitivity to the initial conditions, popularly known as "butterfly effect". This random position is initialized with the default PRNG, as a one-off randomizer. Once the start position of the chaotic system has been obtained, the system generates the next sequence using its current position. Used approach is based on the following definition (3):

$$rndreal = \mathrm{mod}(\mathrm{abs}(rndChaos), 1.0) \tag{3}$$

6 Experiment Design

For the purpose of ChaosDE performance comparison within this research, the Schwefel's test function (4), shifted Grienwang function (5), and shifted Ackley's original function in the form (6) and shifted Rastrigin's function (7) were selected.

$$f(x) = \sum_{i=1}^{\dim} -x_i \sin\left(\sqrt{|x_i|}\right) \tag{4}$$

Function minimum:
Position for $\mathrm{E_n}$: $(x_1, x_2 \ldots x_\mathrm{n}) = (420.969,\ 420.969, \ldots,\ 420.969)$
Value for $\mathrm{E_n}$: $y = -418.983 \cdot dim$; Function interval: $< -500,\ 500 >$.

$$f(x) = \sum_{i=1}^{\dim} \frac{(x_i - s_i)^2}{4000} - \prod_{i=1}^{\dim} \cos(\frac{x_i - s_i}{\sqrt{i}}) + 1 \tag{5}$$

Function minimum: Position for $\mathrm{E_n}$: $(x_1, x_2 \ldots x_\mathrm{n}) = \boldsymbol{s}$; Value for $\mathrm{E_n}$: $y = 0$
Function interval: $< -50,\ 50 >$.

$$f(x) = -20\exp\left(-0.02\sqrt{\frac{1}{D}\sum_{i=1}^{\dim}(x_i - s_i)^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{\dim}\cos 2\pi(x_i - s_i)\right) + 20 + \exp(1) \tag{6}$$

Function minimum: Position for $\mathrm{E_n}$: $(x_1, x_2 \ldots x_\mathrm{n}) = \boldsymbol{s}$; Value for $\mathrm{E_n}$: $y = 0$
Function interval: $< -30,\ 30 >$.

$$f(x) = 10\dim\sum_{i=1}^{\dim}(x_i - s_i)^2 - 10\cos(2\pi x_i - s_i) \tag{7}$$

Function minimum: Position for $\mathrm{E_n} : (x_1, x_2 \ldots x_\mathrm{n}) = \boldsymbol{s}$, Value for $\mathrm{E_n} : y = -90000(\dim 30)$

Function interval: $< -5.12,\ 5.12 >$.

Where s_i is a random number from the 90% range of function interval; $\boldsymbol{s}$ vector is randomly generated before each run of the optimization process.

The parameter settings for both canonical DE and ChaosDE were obtained based on numerous experiments and simulations (see Table 2). It was experimentally determined, that ChaosDE requires lower values of *Cr* parameter [25] for any type of used CPRNG. Canonical DE is using the recommended settings [1]. The maximum number of generations was fixed at 1500 generations. This allowed the possibility to analyze the progress of DE within a limited number of generations and cost function evaluations. Experiments were performed in the environment of *Wolfram Mathematica*; canonical DE therefore has used the built-in *Wolfram Mathematica* pseudo random number generator *Wolfram Cellular Automata* representing traditional pseudorandom

Table 2. Parameter set up for ChaosDE and Canonical DE

DE parameter	Value
Popsize	75
F (for ChaosDE)	0.4
CR (for ChaosDE)	0.4
F (for Canonical DE)	0.5
CR (for Canonical DE)	0.9
Dim	30
Max. Generations	1500

number generator in comparisons. All experiments used different initialization, i.e. different initial population was generated within the each run of Canonical or ChaosDE.

7 Results

Statistical results for the Cost Function (CF) values are shown in comprehensive Tables 3–6 for all 50 repeated runs of DE/ChaosDE, four different benchmark functions and five randomization schemes.

Table 3. Simple results statistics for the Canonical DE and ChaosDE; Schwefel's function

DE version	Avg CF	Median CF	Max CF	Min CF	StdDev	p-value
Canonical DE	−5493.26	−5339.34	−4944.96	−6628.4	**440.8144**	–
Burgers dist	−10375.9	−10360	**−9245.86**	−11722.9	518.8032	0.010864
Burgers map	**−10793.5**	**−11413.9**	−6787.51	−12328.1	1387.362	
Lozi dist	−8709.74	−8530.74	−7814.36	−11042.5	661.7437	0.000112
Lozi map	−9932.46	−9922.25	−8200.06	**−12530.9**	1043.777	

The bold values within the all Tables 3–6 depict the best obtained results, italic values are considered to be similar. Statistical comparisons are based on the *Wilcoxon signed-rank test* with significance level 0.05; and performed for the pairs of ChaosDE with CPRNG and identified similar PRNG distribution. The graphical comparisons of the time evolution of average CF values for all 50 runs of five versions of DE/ChaosDE with different randomizations and two selected benchmark functions are depicted in Figs. 3 and 4. The notation in Tables and Figures is following: *Burgers/Lozi Map* represents the chaotic based CPRNG, whereas *Burgers/Lozi Dist* represents identified distribution PRNG.

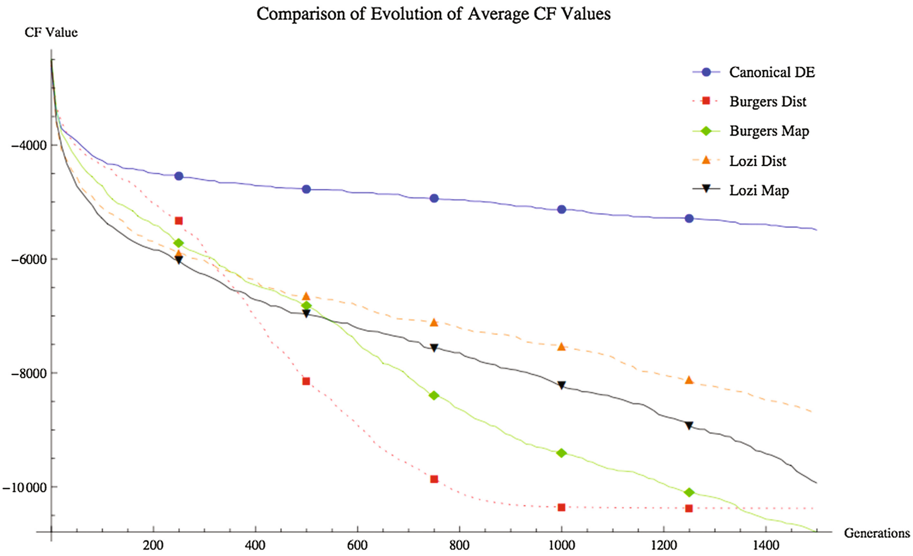


Fig. 3. Comparison of the time evolution of avg. CF values for the all 50 runs of Canonical DE, and four versions of ChaosDE with different randomization. Schwefel's function.

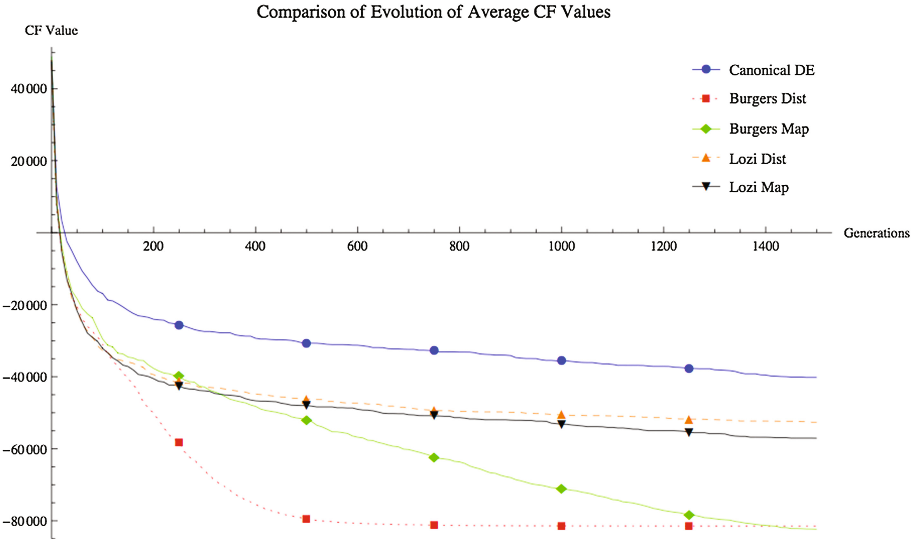


Fig. 4. Comparison of the time evolution of avg. CF values for the all 50 runs of Canonical DE, and four versions of ChaosDE with different randomizations. Shifted Rastrigin's function.

Table 4. Simple results statistics for the Canonical DE and ChaosDE; shifted Rastrigin's func.

DE version	Avg CF	Median CF	Max CF	Min CF	StdDev	p-value
Canonical DE	−40188.49	−39264.95	−33629.64	−49994.34	3983.79	–
Burgers dist	−81465.42	−82256.88	**−74959.67**	−85542.21	2521.65	0.02812
Burgers map	**−82339.03**	**−83552.39**	−63945.49	**−87977.11**	5262.81	
Lozi dist	−52641.81	−52722.69	−49731.38	−57271.93	**1851.70**	$8.0695.10^{-6}$
Lozi map	−57054.66	−56667.65	−52235.56	−62969.12	2927.69	

Table 5. Simple results statistics for the Canonical DE and ChaosDE; shifted Ackley's func.

DE version	Avg CF	Median CF	Max CF	Min CF	StdDev	p-value
Canonical DE	3.38E-09	2.68E-09	7.55E-09	9.48E-10	1.74E-09	–
Burgers dist	4.333288	4.554203	7.464985	2.013873	1.328967	$1.8253.10^{-6}$
Burgers map	1.43E-06	1.25E-12	1.775137	1.47E-14	0.391209	
Lozi dist	1.64E-14	***1.47E-14***	3.6E-14	***7.55E-15***	5.26E-15	0.5231
Lozi map	**1.54E-14**	***1.47E-14***	**2.89E-14**	***7.55E-15***	**4.11E-15**	

Table 6. Simple results statistics for the Canonical DE and ChaosDE; shifted Grienwang func.

DE version	Avg CF	Median CF	Max CF	Min CF	StdDev	p-value
Canonical DE	***0***	***0***	***0***	***0***	***0***	–
Burgers dist	0.525982	0.514041	0.998373	0.098319	0.26012	$1.8626.10^{-6}$
Burgers map	6.89E-07	1.47E-09	0.15187	0	0.00323	
Lozi dist	***0***	***0***	***0***	***0***	***0***	1
Lozi map	***0***	***0***	***0***	***0***	***0***	

8 Conclusions

The primary aim of this work is to experimentally investigate the utilization of the various discrete chaotic systems, as the chaotic pseudo random number generator embedded into DE. Experiments are focused on the extended investigation, whether the different randomization and pseudo random numbers distribution given by particular PRNG or hidden complex chaotic dynamics providing the unique sequencing are more beneficial to the heuristic performance. The findings can be summarized as:

- Obtained graphical comparisons and data in Tables 3–6 and Figs. 3 and 4 support the claim that chaos driven heuristic is more sensitive to the hidden chaotic dynamics driving the selection, mutation, crossover or other processes through CPRNG. The influence of different PRNG randomization (distribution) type is strengthened by the presence of chaotic dynamics and sequencing in the pseudo random series given by the dynamics of discretized chaotic attractor/flow.
- It is clear that (selection of) the best CPRNGs are problem-dependent. By keeping the information about the chaotic dynamics driving the selection/mutation processes

inside heuristic, its performance is significantly different: either better or worse against other compared versions.

- In the first two cases (Schwefel and shifted Rastrigin function – Tables 3 and 4), the performance of ChaosDE was significantly better in comparison with canonical DE. Furthermore the effect of different PRNG distribution became even stronger with the chaotic dynamics kept inside CPRNG sequences. Lozi map based CPRNG has given stable better performance than similar identified PRNG. Both Lozi map based PRNG/CPRNG have been outperformed by the utilization of Burgers map based PRNG/CPRNG. An interesting phenomenon has been revealed. The Burgers map based not-chaotic PRNG drives DE to the strong and fast progress towards function extreme (local) followed by premature population stagnation phase. Whereas Burgers map CPRNG with chaotic dynamics secured the continuous development of population towards global best solution without stagnation.
- The third and the fourth case study (Tables 5 and 6) have given absolutely reversed character of results. Performance of Lozi based CPRNG/PRNG is comparable even with canonical DE (slightly better results for Lozi map CPRNG and Ackley function), whereas the Burgers map based randomization has given worse results. As aforementioned in the previous point, the premature stagnation for PRNG has occurred also here (more considerable), whereas the Burgers map based CPRNG with chaotic dynamics has driven the DE more or less towards the function extreme.
- Since the aim was to investigate the randomization/sequencing of indices selection inside DE, only the simplest canonical *DE/Rand/1/Bin* strategy has been utilized in this research. The parameter adjustment/strategy adaptation or ensembles techniques in jDE, EPSDE, SHADE may significantly interact with the dynamics of sequencing (selection) of indices driven by particular not-uniform PRNG/CPRNG.
- Sequencing of pseudo random numbers and chaotic dynamics hidden inside pseudo random series can be significantly changed by the selection of chaotic systems, thus to avoid the CF landscape dependency. The simplest way for changing the influence to the heuristic during the run is to swap currently used chaotic system for different one, or to change the internal parameters of chaotic systems (Table 1).
- Furthermore many previous implementations of chaotic dynamics into the evolutionary/swarm based algorithms (not-adaptive/adaptive/ensemble based) showed that it is advantageous, since it can be easily implemented into any existing algorithm as a plug-in module.

Acknowledgements. This work was supported by Grant Agency of the Czech Republic - GACR P103/15/06700S, further by financial support of research project NPU I No. MSMT-7778/2014 by the Ministry of Education of the Czech Republic and also by the European Regional Development Fund under the Project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089, and by Internal Grant Agency of Tomas Bata University under the projects No. IGA/CEBIA-Tech/2017/004.

References

1. Price, K.V., An introduction to differential evolution. In: Corne, D., Dorigo, M., Glover, F., (eds.) New Ideas in Optimization, pp. 79–108. McGraw-Hill Ltd., London (1999)
2. Brest, J., Greiner, S., Boskovic, B., Mernik, M., Zumer, V.: Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. IEEE Trans. Evol. Comput. **10**(6), 646–657 (2006)
3. 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)
4. Das, S., Suganthan, P.N.: Differential evolution: a survey of the state-of-the-art. IEEE Trans. Evol. Comput. **15**(1), 4–31 (2011)
5. Das, S., Mullick, S.S., Suganthan, P.N.: Recent advances in differential evolution – an updated survey. Swarm Evol. Comput. **27**, 1–30 (2016)
6. Neri, F., Tirronen, V.: Recent advances in differential evolution: a survey and experimental analysis. Artif. Intell. Rev. **33**(1–2), 61–106 (2010)
7. Weber, M., Neri, F., Tirronen, V.: A study on scale factor in distributed differential evolution. Inf. Sci. **181**(12), 2488–2511 (2011)
8. Neri, F., Iacca, G., Mininno, E.: Disturbed exploitation compact differential evolution for limited memory optimization problems. Inf. Sci. **181**(12), 2469–2487 (2011)
9. Iacca, G., Caraffini, F., Neri, F.: Compact differential evolution light: high performance despite limited memory requirement and modest computational overhead. J. Comput. Sci. Technol. **27**(5), 1056–1076 (2012)
10. Zamuda, A., Brest, J.: Self-adaptive control parameters' randomization frequency and propagations in differential evolution. Swarm Evol. Comput. **25**, 72–99 (2015)
11. Caponetto, R., Fortuna, L., Fazzino, S., Xibilia, M.G.: Chaotic sequences to improve the performance of evolutionary algorithms. IEEE Trans. Evol. Comput. **7**(3), 289–304 (2003)
12. Davendra, D., Zelinka, I., Senkerik, R.: Chaos driven evolutionary algorithms for the task of PID control. Comput. Math. Appl. **60**(4), 1088–1104 (2010)
13. Zelinka, I.: SOMA — self-organizing migrating algorithm. In: Zelinka, I. (ed.) New Optimization Techniques in Engineering. Studies in Fuzziness and Soft Computing, vol. 141, pp. 167–217. Springer, Heidelberg (2004)
14. dos Santos Coelho, L., Mariani, V.C.: A novel chaotic particle swarm optimization approach using Hénon map and implicit filtering local search for economic load dispatch. Chaos, Solitons Fractals **39**(2), 510–518 (2009)
15. Pluhacek, M., Senkerik, R., Davendra, D., Kominkova Oplatkova, Z., Zelinka, I.: On the behavior and performance of chaos driven PSO algorithm with inertia weight. Comput. Math. Appl. **66**(2), 122–134 (2013)
16. Pluhacek, M., Senkerik, R., Zelinka, I., Davendra, D.: Chaos PSO algorithm driven alternately by two different chaotic maps – an initial study. In: 2013 IEEE Congress on Evolutionary Computation (CEC), 20–23 June 2013, pp. 2444–2449 (2013)
17. Pluhacek, M., Senkerik, R., Davendra, D.: Chaos particle swarm optimization with Eensemble of chaotic systems. Swarm Evol. Comput. **25**, 29–35 (2015)
18. Metlicka, M., Davendra, D.: Chaos driven discrete artificial bee algorithm for location and assignment optimisation problems. Swarm Evol. Comput. **25**, 15–28 (2015)
19. dos Santos Coelho, L., Ayala, H.V.H., Mariani, V.C.: A self-adaptive chaotic differential evolution algorithm using gamma distribution for unconstrained global optimization. Appl. Math. Comput. **234**, 452–459 (2014)

20. Senkerik, R., Pluhacek, M., Zelinka, I., Davendra, D., Janostik, J.: Preliminary study on the randomization and sequencing for the chaos embedded heuristic. In: Abraham, A., Wegrzyn-Wolska, K., Hassanien, A.E., Snasel, V., Alimi, A.M. (eds.) Proceedings of the Second International Afro-European Conference for Industrial Advancement AECIA 2015. AISC, vol. 427, pp. 591–601. Springer, Cham (2016). doi:10.1007/978-3-319-29504-6_55
21. Zelinka, I.: A survey on evolutionary algorithms dynamics and its complexity – mutual relations, past, present and future. Swarm Evol. Comput. **25**, 2–14 (2015)
22. 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)
23. Price, K.V., Storn, R.M., Lampinen, J.A.: Differential Evolution – A Practical Approach to Global Optimization. Natural Computing Series. Springer, Heidelberg (2005)
24. Sprott, J.C.: Chaos and Time-Series Analysis. Oxford University Press, New York (2003)
25. Senkerik, R., Pluhacek, M., Kominkova Oplatkova, Z., Davendra, D.: On the parameter settings for the chaotic dynamics embedded differential evolution. In: 2015 IEEE Congress on Evolutionary Computation (CEC), 25–28 May 2015, pp. 1410–1417 (2015)

Author Index

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1