Editor **DANA SIMIAN**

MODELLING AND DEVELOPMENT OF INTELLIGENT SYSTEMS

Proceedings of the Fifth International Conference on
MODELLING AND DEVELOPMENT OF INTELLIGENT SYSTEMS

23 - 25 June, 2017, Sibiu, ROMANIA

Lucian Blaga University Press, 2018

Editor DANA SIMIAN

MODELLING AND DEVELOPMENT OF INTELLIGENT SYSTEMS

Proceeding of the Fifth International Conference
"Modelling and Development of Intelligent Systems"
June 23 - 25, 2017, Sibiu, ROMANIA

"Lucian Blaga" University Press, Sibiu

2018

“Lucian Blaga” University Press, Sibiu, 2018

Editor Dana Simian

All papers in this volume were peer review by two independent reviewers

ISSN 2067-3965

Associate editor Laura Florentina Stoica
Cover design Ralf Fabian

Proceedings of the Fifth International Conference
Modelling and Development of Intelligent Systems
June 23 - 25, 2017, Sibiu, ROMANIA

Copyright @ 2018 All rights reserved to editors and authors

Preface

This volume contains refereed papers presented at the 5th International Conference on Modelling and Development of Intelligent Systems which was held between June 22-25, 2017, at the Faculty of Sciences, "Lucian Blaga" University of Sibiu, Romania. The MDIS conference provides an opportunity for sharing ideas and establishing scientific cooperation in the field of intelligent systems. It aims to bring together scientists, researchers, students, interested and working in fields which can be connected with modeling and development of intelligent systems. Specific topics of the conference includes but are not restricted to: evolutionary algorithms, evolutionary computing, genetic algorithms and their applications, modelling and optimization of dynamic systems, adaptive systems, multiagent systems, swarm intelligence, metaheuristics and applications, machine learning, self learning algorithms, intelligent systems for decision support, knowledge based systems, data mining, ontology engineering, pattern recognition and model checking, motion recognition, e-learning, hybrid computation for artificial vision, knowledge reasoning for artificial vision, geometric modelling and spatial reasonings, grid computing and clustering, large scale optimization techniques and mathematical models for development of intelligent systems.

The plenary invited speakers addressed current topics in the conference field:

- Detlef Streitferdt, "Decision Model for a Cycle Computer Developers Environment**"**

- Milan Tuba, **"**Stochastic Optimization for Classification Algorithms"

The talks were presented by scientists, researchers and students from 10 countries (Bosnia and Herzegovina, Croatia, Germany, Norway, Portugal, Qatar, Romania, Serbia, Switzerland, UAE). During the conference a wide range of theoretical and practical problems related to the conference topics were discussed.

We thank all the participants for their interesting talks and discussions. We also thank the members of the scientific committee for their help in reviewing the submitted papers and for their contributions to the scientific success of the conference and to the quality of this proceedings volume.

June 2018

Dana Simian
Conference chairman

Scientific Committee

Kiril Alexiev - Bulgarian Academy of Sciences, Bulgaria
Charul Bhatnagar - Institute of Engineering and Technology, GLA University – India
Alina Barbulescu - Higher College of Technology, Sharjah, UAE
Lasse Berntzen, Buskerud and Vestfold University College, Norway
Florian Boian - Babes-Bolyai University of Cluj Napoca, Romania
Steve Cassidy - Macquarie University, Australia
Dan Cristea - "Alexandru Ioan Cuza" University of Iasi, Romania
Gabriela Czibula - Babes-Bolyai University of Cluj Napoca, Romania
Thierry Declerck - German Research Centre for Artificial Intelligence, DFKI, Kaiserslautern, Germany - Austrian Center for Digital Humanities & DFKI GmbH, Austria
Alexiei Dingli - University of Malta, Malta
Oleksandr Dorokhov - Kharkiv National University of Economics, Ukraine
George Eleftherakis - University of Sheffield England - The University of Sheffield International Faculty, City College Thessaloniki, Greece
Ralf Fabian - "Lucian Blaga" University of Sibiu, Romania
Stefka Fidanova - IICT-BAS, Bulgarian Academy of Sciences, Sofia, Bulgaria
Ulrich Fiedler, Bern University of Applied Science, Switzerland
Martin Fränzle - Carl von Ossietzky Universität Oldenburg, Germany
Amir Gandomi, Michigan State University, USA
Andrina Granić - University of Split, Croatia
Dejan Gjorgjevikj - Ss. Cyril and Methodius University - Skopje, Republic of Macedonia
Katalina Grigorova - University of Ruse, Bulgaria
Axel Hahn, Carl von Ossietzky Universität, Oldenburg, Germany
Masafumi Hagiwara - Keio University, Japan
Raka Jovanovic, Hamad bin Khalifa University, Qatar
Adnan Khashman - European Centre for Research and Academic Affairs (ECRAA), Lefkosa, Nicosia, Cyprus
Wolfgang Kössler - Humboldt-Universität Berlin, Germany
Suzana Loskovska - "Ss. Cyril and Methodius" University, Republic of Macedonia in Skopje, Republic of Macedonia
Manuel Campos Martinez - Universidad de Murcia, Spain
Gines Garcia Mateos - Universidad de Murcia, Spain
Gerard de Melo - Rutgers, The State University of New Jersey, USA
Matthew Montebello - University of Malta, Malta
G. Jose Moses - Raghu Engineering College Visakhapatnam, Andhra Pradesh, India
Eugénio Costa Oliveira - University of Porto, Portugal
Anca Ralescu - University of Cincinnati, United States of America

Willi Sauerbrei - University of Freiburg, Germany
Abdel-Badeeh M. Salem - Ain Shams University, Cairo, Egypt
Klaus Bruno Schebesch - "Vasile Goldis" University, Arad, Romania
Vasile-Marian Scuturici - Université de Lyon, France
Livia Sangeorzan - Transilvania University, Brașov, Romania
Francesco Sicurello - University of Milano Bicocca, President of IITM, Italy
Dana Simian - "Lucian Blaga" University of Sibiu, Romania
Ansgar Steland - RWTH Aachen University, Germany
Florin Stoica - "Lucian Blaga" University of Sibiu, Romania
Detlef Streitferdt - Ilmenau University of Technology, Software Architectures and Product Lines Group, Germany
Ying Tan, Beijing University, China
Claude Touzet - Aix Marseille Université, Lab. de Neurosciences Intégratives et Adaptatives, France
Milan Tuba - John Naisbitt University of Belgrade, Serbia
Dan Tufis - Romanian Academy, Research Institute for Artificial Intelligence "Mihai Draganescu", Bucharest, Romania
Anca Vasilescu - Transilvania University, Brașov, Romania
Xin-She Yang, Middlesex University London, UK

Table of Contents

Plenary Lecture - Decision Model for a Cycle Computer Developers Environment
Detlef Streitferdt 9

Plenary Lecture - Stochastic Optimization for Classification Algorithms
Milan Tuba ……..……..……..……..……..……..……..…. 11

The integration of augmented reality in everyday life
Petrică Bota, Robert Săndică, Dana Simian …….....…..... 13

An Imperialist Competitive Algorithm Optimized to Solve the Travelling Salesman Problem
Stelian Ciurea ……..…..……...…….. 20

Why and how Practice impacts Confidence in introductory object oriented programming Courses
Viggo Holmstedt, Shegaw Anagaw Mengiste …… 29

Using Top Down Clustering model for the assessment of anthropogenic pressures and threats on the fish populations
Daniel Hunyadi, Angela Bănăduc, Doru Bănăduc …….……...…..….... 43

Factors Influencing Group Creativity of Computer Science Students: A Fuzzy Cognitive Map
Gabriela Moise, Monica Vlădoiu, Zoran Constantinescu ……...…..…..… 50

Improving of e-business activities by building web applications with integrated data mining services
Mircea Adrian Mușan, Iuliana Maria Cândea …….……..….. 60

Theoretical and practical approaches for documents classification
Livia Sângeorzan, Nicoleta Enache-David …….....….. 67

Tuning Extreme Learning Machines with genetic algorithms
Florin Stoica, Alina Bărbulescu, Laura Florentina Stoica …….....……. 72

Moth Search Algorithm for Bound Constrained Optimization Problems
Eva Tuba, Romana Capor-Hrosik, Adis Alihodzic, Marko Beko, Raka Jovanovic 82

Author index……. 90

Fifth International Conference
Modelling and Development of Intelligent Systems
June 23 - 25, 2017
"Lucian Blaga" University
Sibiu - Romania

Plenary Lecture

Decision Model for a Cycle Computer Developers Environment

Detlef Streitferdt

Department of Computer Science and Automation
Technische Universität Ilmenau
Ilmenau, GERMANY
E-mail: detlef.streitferdt@tu-ilmenau.de

Current software development efforts are required to address very short development cycles for complex systems with high demands on the quality of the resulting product. At the same time the environments for software developers are manifold and are getting more and more complex as well, what requires relevant additional efforts to setup and maintain developers environments. Development methods and processes as well as the required and corresponding tools (to the development steps) are part of such developers environments.

Within a university project our research group developed a Cycle Computer. The developers environment for this project is based upon a large number of decisions for an optimized development process and tool set. This contribution introduces and explains the details of the Cycle Computer projects developers environment.

The ontology decision model is based on the PhD work of Franz Füßl with five ab- straction levels to capture and maintain constraints, interconnect them and use the model for automated decisions using deduction and ontology learning. This models extends trope- based (existentially dependent elements) ontologies by the inclusion of arbitrary metrics (e.g., based on measurements) and social factors.

At the most abstract 4th level the model hosts data sources representing very simple issues which are captured with corresponding multiple-choice or single-choice questions. Data sources may also use sensor values or measurements. The 3rd level includes the fea-tures of a project (e. g. budget, operating system or personal motivation). The features are connected to at least one data source element. Each feature is measured on a nominal, ordinal or metric scale which also corresponds to the connected data source element type. Features are connected to the cells on the 2nd level. Cells generate knowledge based on the connected features. The 1st level hosts items (e. g., requirements engineering, software architecture pattern) which use the information stored in the cells to model abstract compo- nents for the solution. Finally, solutions are at level zero and represent developer packets to be used in a given development effort. The selection of the solution packets is based on their feasibility to fulfill the items of the first level. The knowledge model is a directed graph. Arbitrary associations can be realized in this graph. Currently five associations have been defined (is path, has path, can path, part-of path, and used-for path) and fully realized in a software tool.

The Cycle Computer is a project for students up to PhD level. Embedded components, Android Apps and Windows (C++, C#) components are integrated into it. For such a large project

many different tools and development processes need to be interconnected seam- lessly. With our ontology-based decision model the student ideas and preferences for the tool and process landscape can be captured, modeled and used to reason about specific components of developers environments.

At the data source level questions regarding the team roles, the experience level with technologies like Android, the MSP430 microcontroller or Bluetooth are asked, together with questions about the motivation or interdisciplinary knowledge are asked. Thus, a team specific view can be built with such questions. The results are used on the following feature level. Here, results are generated based on the given answers. The cells of the next level have been used to represent the possible answers for a feature with "isn't-it" relations and the feature itself uses the "has" relation to the corresponding questions. On the item level concepts like the "development method" as a whole or "requirements engineering" as part of the method are modeled. To model the knowledge and interconnect it to the answers of the students, items are connected via further cells to the cell – feature – data source path. As an example the feature "Scrum" is a "development method" and it needs the "feasibility" feature what cannot be fullfilled by "undisciplined" teams.

The deduction process is supported by algorithms built into the model. Currently, the most complex query, the "find"-algorithm is used to find tools compatible with the team situation. The situation was given by answering the questions. Now we can ask questions like "Find elements to prioritize requirements". The resulting tool is the best / optimized fit according to the above set of answers. To cover the continuously changing body of knowledge the ontology model is able to learn by weights at the edges of the model. The weights can then be adapted / adjusted by e.g. inductive reasoning.

Based on the Cycle Computer developers environment the improved acceptance of the individually "selected" developers environment can be shown and explained. The selection of a developers environment can be traced back to ontology based decisions, the knowledge model. The future goal is the further automation of the selection process for complete developers environments.

Brief Biography of the Speaker: Detlef Streitferdt is currently senior researcher at the Ilmenau University of Technology heading the research group Software Architectures and Product Lines since 2010. The research fields are the efficient development of software architectures and product lines, their analysis and their assessment as well as software development processes and model-driven development. Before returning to the University he was Principal Scientist at the ABB AG Corporate Research Center in Ladenburg, Germany. He was working in the field of software development for embedded systems. Detlef studied Computer Science at the University of Stuttgart and spent a year of his studies at the University of Waterloo in Canada. He received his doctoral degree from the Technical University of Ilmenau in the field of requirements engineering for product line software development in 2004.

Fifth International Conference
Modelling and Development of Intelligent Systems
June 23 - 25, 2017
"Lucian Blaga" University
Sibiu - Romania

Plenary Lecture

Stochastic Optimization for Classification Algorithms

Milan Tuba

Faculty of Computer Science
John Naisbitt University
Belgrade, SERBIA
E-mail: tuba@matf.bg.ac.rs

Classification problem is an important research topic in computer science. It is used in data mining or machine learning to detect patterns in the input data and to determine what class each instance belongs to. Applications of classification are numerous and include different areas like medicine for tumor and diseases classification, image processing, economy for stock trend forecasting, ecology for agricultural, forest, plant classification, etc. Classification belongs to supervised machine learning where instances are given with corresponding labels (classes). Some of the most important supervised machine learning techniques are based on artificial intelligence, perception-based techniques and statistics. Some of the supervised learning algorithms used for classifications are decision trees, logistic regression, artificial neural networks, k-nearest neighbors, etc. Support vector machine (SVM) is one of the latest and most efficient supervised machine learning algorithms and it has been successfully used for many different classification problems. SVM determines a hyperplane that separates data from different classes. It first builds a model based on instances from the training set and then uses that model for further classification of unknown instances. Real world data are practically never perfectly separable so a soft margin parameter that affects the trade-off between complexity of the model and proportion of non-separable samples was introduced in the SVM model. Additionally, in order to adjust SVM for classification of non-linearly separable data, projection to higher-dimension space by kernel function was introduced. Gaussian radial basis function is the most common used kernel function and its parameter defines the influence of a single training example to the model. The successfulness of the SVM model depends on the soft-margin coefficient as well as on the parameter of the kernel function hence selecting optimal values for these parameters is a crucial step in SVM construction. One of the most used techniques for SVM parameter tuning is grid search on the log-scale of the parameters, combined with cross validation procedure. This technique may result in huge computational time and far from optimal selection of parameters. Selecting a good pair of values for parameters is a hard optimization problem and for such problems, stochastic population search algorithms, particularly swarm intelligence, were studied and used. In this plenary lecture some recent successful applications of the swarm intelligence algorithms to support vector machine parameters optimization will be presented.

Brief Biography of the Speaker: Milan Tuba is the Dean of Graduate School of Computer Science and Provost for mathematical and technical sciences at John Naisbitt University of Belgrade. He received B. S. in Mathematics, M. S. in Mathematics, M. S. in Computer Science, M. Ph. in Computer Science, Ph. D. in Computer Science from University of Belgrade and New York University. From 1983 to 1994 he was in the U.S.A. first as a graduate student and teaching and research assistant at Vanderbilt University in Nashville and

Courant Institute of Mathematical Sciences, New York University and later as Assistant Professor of Electrical Engineering at Cooper Union School of Engineering, New York. During that time he was the founder and director of Microprocessor Lab and VLSI Lab, leader of scientific projects and theses supervisor. From 1994 he was Assistant Professor of Computer Science and Director of Computer Center at University of Belgrade, from 2001 Associate Professor, Faculty of Mathematics, University of Belgrade, and from 2004 also a Professor of Computer Science and Dean of the College of Computer Science, Megatrend University Belgrade. He was teaching more than 20 graduate and undergraduate courses, from VLSI Design and Computer Architecture to Computer Networks, Operating Systems, Image Processing, Calculus and Queuing Theory. His research interest includes heuristic optimizations applied to computer networks, image processing and combinatorial problems. Prof. Tuba is the author or coauthor of more than 150 scientific papers and coeditor or member of the editorial board or scientific committee of number of scientific journals and conferences. Member of the ACM, IEEE, AMS, SIAM, IFNA.

Fifth International Conference
Modelling and Development of Intelligent Systems
June 23 - 25, 2017
"Lucian Blaga" University
Sibiu - Romania

The integration of augmented reality in everyday life

Petrică Bota, Robert Săndică, Dana Simian

Abstract

Ever since their invention computers were designed to improve the user's performance and aid them in solving diverse tasks. Since the launch of the smartphone computers occupy a larger role in our lives and have become an extension of the user. The available interfaces that connect users to the computers and the Internet are slow in comparison to the processing power of the available computers and can be improved upon. The forms of input are limited to writing, pushing buttons, clicking or touchscreen. The goal of this article is to propose a new solution to the issues of data input and output. Our solution is to provide real-time augmented reality elements through the use of devices such as smartphones or augmented reality glasses. The purpose of these devices is to receive data without the user having to input it manually and offering information and assistance through the representation of 3D elements. As proof of concept we have implemented an Android app that helps the user reach his desired destination by using overlapping virtual cues on top of the camera's video stream. We have found that it is much easier to navigate with the help of the virtual signs due to the fact they provide clear, concise and difficult to misunderstand information. Though, we have developed this app as a proof of concept we consider that augmented reality can be implemented, in one form or another, into multiple fields such as: medicine, teaching, military, etc.

1 Augmented reality. Introduction

Processing power has drastically increased since the invention of computers but the way in which we interact with a computer hasn't changed much in the past decades. We consider that the main issue in working with computers is not the processing power of the computer but the speed with which we introduce and retrieve information. For instance, the IBM 7030 ("Stretch") released in 1961was able to perform 1.200 MIPS (millions of instructions per second) [1] at 3.30 MHz while more recently the AMD Ryzen 7 1800X released in 2017 is able to perform 304,510 MIPS at 3.6 GHz [2]. We can observe that the performance upgrade in processing power over the last 50 years is in the order of thousands of millions. Meanwhile, the average rate of introducing data with the use of a keyboard is around 50 wpm (words per minute). There are also exceptional individuals, such as Stella Pajunas or Barbara Blackburn, who can exceed 200 wpm[3]. The processing power of a computer is much faster than a person's ability to input or interpret data. For inputting data we are limited to

pushing buttons, touching screens, moving a mouse or a controller. For interpreting data we are limited by the size of our screens, our ability to interpret sounds and the speed of our reading.

The issue of user input/interpreting speed could be resolved either by implementing a neural net headset [4] or by adding an additional layer of virtual information on top of the real world. A neural net headset is a device that in theory can directly transmit the user's brainwaves to a computer. Elon Musk is one of the supporters of the neural net headset concept, especially since he started the company "Neuralink"[5] with the purpose of developing viable devices for direct human-computer communication. Another way to solve the issue of data inputting and data interpreting can be augmented reality (AR). With the help of AR we can add another layer of information between our eyes and the real world. The additional layer of information can be used for data input and output

The goal of this paper is to study how AR applications can be used in order to improve the speed of data communication between humans and computers.

The main reasons we chose to focus on the implementation of augmented reality and not on a neural net headset are:

- The technology required to interpret brain signals is still being developed;
- A neural net head set would probably be considered a rather invasive device;
- The technology required for viable augmented reality exists and it is already being implemented by multiple companies such as Oculus, Vive, Microsoft;
- Any modern smartphone is capable of augmented reality features;
- A wearable such as AR (augmented reality) glasses is far less invasive than a wearable that is directly connected to your brain impulses; therefore the social taboo shouldn't be as relevant as in the case of the neural net.

An AR app works by projecting virtual information on a virtual layer that is displayed between the user's eyes and real life objects. In that virtual layer it is possible to display additional information regarding real life objects or interactions. Augmented reality has gained popularity over the past years but the technology itself is several decades old. The first AR system that provided and immersive experience by combining real life objects and virtually generated ones was developed in 1992 at the U.S. Air Force's Armstrong Labs in 1992 [6]. The device projects additional information on the pilot's helmet such as longitude, latitude or automatically colors enemy targets.

Augmented reality has reached millions of people through the smartphone. The smartphone is a device that is able to record and display information simultaneously; that makes it is perfect for the use of AR apps. Also, it is much cheaper than the devices used by the military.

An example of AR app is Pokemon Go. This AR app was released in 2016. It is the most popular AR app with more than 60 million active user as of 2017 [7]. The huge number of people who already use AR apps prove that users are interested in this technology.

At the moment, the most common devices that can be used to run AR apps are smartphones and AR glasses such as Google Glass [8] or HoloLens [9].

There are two main types of AR apps:

a) Apps that use marker points as reference for adding the virtual information;
b) Apps that don't require markers points in order to add data.

Marker based AR apps requires that you pre-program elements that will be recognized by the app which will then respond accordingly, either by displaying additional information or starting another process. For example, we could have a 2D picture of a pyramid that is recognized by the app which then adds a 3D pyramid model on top of the 2D image. It is much easier to program a marker

based AR app due to the fact that you simply program how the app should respond when it encounters a scenario for which it was prepared.

The second type of AR app does not required markers in order to function. This type of AR app should be able to identify patterns, colors or some other features that may exist in a camera frame and react accordingly, without being explicitly programmed for each specific scenario.

The main aim of this article is to study how AR apps could be used to improve the transmission of information to users. For this purpose we realized a comparison between the two types of AR apps. To highlight the advantages and disadvantages of marker based and markerless AR apps we developed an app of each type. We consider that this constructive method is able to cover all aspects related to our goal: app design, technologies available, evaluation of results.

The first app that we developed in order to study the AR technology was Virtual Reader[10]. Virtual Reader is a marker based AR app that provides additional information to the user by adding 3D elements on top of a predetermined 2D image. The next app we developed was a markerless AR app named Virtual Pathfinder[11]. The purpose of this app is to provide additional information that aids user to get from his current location to another one that he selected. This was done by adding a virtual path on top of a smartphone's video feed.

The rest of the article is organized as follows. In Section 2 we present the two apps that we developed and their capabilities. This section contains our original results. Section 3 presents the conclusions of this article. It shows how our apps relates to the initial goal of this article.

2 AR apps

Both apps we developed require specific devices and technologies. We took into consideration the most affordable devices for users. Smartphones are the baseline for the development of our AR apps. Smartphones are wide-spread. A report done by Wireless Smartphone Strategies (WSS) services [12] that encompassed 88 countries reveals the fact that more than 44% of people in developed countries own one smartphone. Moreover, most smartphones are capable of running virtual reality applications.

The implementation process includes apps development and markers design. We chose Unity as a software development environment both for Virtual Reader and Virtual Pathfinder. Unity is compatible with AR technologies and software development and has open marketplaces with lots of free or paid assets [13]. Moreover it contains great built-in physics mechanics that are easy to edit and work with.

For encrypting and decrypting the markers into and from different images we chose Vuforia. Vuforia is compatible with the Unity framework and allows us to create our own markers with the images that we desired. We only used the 2D features that Vuforia offers for AR development but other features like recognizing complex 3D objects or simultaneously recognizing multiple 2D planes of a simple 3D object, such as a cube, are also offered by Vuforia.

2.1 Virtual Reader

We started our research by creating a relatively simple AR app that provides additional information while reading. This information was represented under the form of 3D objects overlapping the 2D predetermined design (Fig. 1).

As mentioned previously a marked based AR app uses certain points as reference in order to "remember" how to react and what information to convey as a response. We used 2D images into which we incorporated these markers that the app uses in order to recognize the image and display the appropriate 3D object. In order to use the app you have to point you smartphone camera at the image that had the marker incorporated. Once the app recognizes the marker it displays the appropriate 3D object.

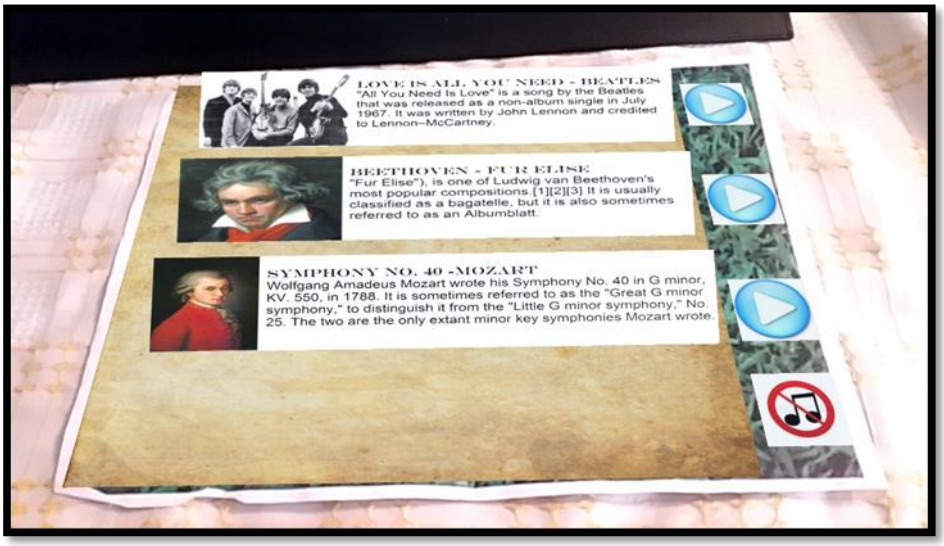


Fig. 1 – screenshot of Virtual Reader

Comparing with other existed similar AR apps, our app includes many features designed specially for improving the interaction with users, enhancing the communication of information. We proposed the following new features:

- Virtual buttons, that can be pushed by inserting your finger or hand between you camera and the 2D model (Fig. 1). The app recognizes the lose of visual contact with the marker specific to the button as a key press event and acts accordingly.

- Virtual photo album with 3D images. The photo album was simulated by adding multiple 3D objects corresponding to a single image. It is possible to swipe forward or through these 3D images by using two virtual buttons.

- Incorporating audio information on top of additional visual information. In our app, one of the marker incorporating images also allows you to play, pause or mute one out of a selection of 3 songs by pushing virtual buttons, but other kind of audio information could be incorporated as well. (Fig. 1).

Using the features enumerated above we are able to add a lot of additional information by using a single image as a base. As an example, we considered a 2D image of Michelangelo as a marker base on top of which we created an entire album of his best works, paintings, and sculptures. The user can browse the volume using virtual buttons. Additional audio information was incorporated, enhancing the user experience and improving the communication of information. This type of display could be used for any 3D object.

The main advantage of the Virtual Reader App when comparing withother AR apps is the fact that the user is able to interact with the virtual environment directly by touch. At the moment, Virtual

Reader touch is limited to pushing virtual buttons; but the level of interaction between the app and the user. can be expanded. Another feature of Virtual Reader is the possibility of transmission of both additional visual and audio information.

For testing the impact of the new proposed features, we presented our app at the event Sibiu Researchers' Night 2016 and 2017 [14]. We discovered that users were more than eager to try and interact with the app. The app was intuitive to use, therefore no prior description or tutorial is required before using Virtual Reader. There were some issues with the identification of the markers if the smartphone was held too tilted. Therefore, a further direction of study is to fix the issue of marker recognition in difficult environments.

2.2 Virtual Pathfinder

We also developed a markerless AR app in order study the advantages and disadvantages of a markerless AR. A markless AR app is much more difficult to develop compared with a marker based AR app because we are not able to pre-program the information which the app should respond to. It is quite difficult to create a general algorithm that will respond to unplanned situations accordingly. Recognizing new scenarios is a task related more to the artificial intelligence field than to the virtual reality one. To overcome this difficulty additional information that substitutes the markers should be taken into account. The most difficult task is to introduce this additional information in the algorithm. In our AR app we use GPS coordinates as additional information.

The purpose of Virtual Pathfinder is to help the user get from his current location to his desired destination. In order to achieve this Virtual Pathfinder uses the smartphone's built in GPS system to draw a route. The main difference from our AR app and a usual navigation app is that the route is represented not only on a 2D plane but on a 3D plane as well. The app differences itself by adding another layer of information on the smartphone's video feed, combining the smartphone's GPS location with the camera video information in order to create 3D virtual cues. These cues are represented either by intermediary signs such as arrows that point what direction you need to follow or by a continuous line that you need to follow in order to reach your destination (Fig. 2).

Fig. 2 – screenshot of Virtual Pathfinder

We encountered multiple technical challenges, out of which most notable were the representation of the representation of the AR elements on to the smartphone screen and the access to maps that would help us navigate the user. We solved this issue representing the 3D images on top of the real world camera feed by inserting another layer/canvas between the smartphone's screen and the video feed. As a result we were able to project our virtual signs on the smartphone's screen indifferent of the position in which the camera was pointed.

Accessing the map information (cities, streets, landmarks) proved to be more challenging than we had estimated. We were unable to access the information from Google Maps in order to implement it in our app due to terms of agreement infringement. Therefore, we personally mapped a small portion of the city (Sibiu) for demonstration purposes.

As far as we know Virtual Pathfinder is the only app that integrates live virtual feedback with GPS mapping. Virtual Pathfinder offers clear and concise information by showing what route the user needs to follow. The route is depicted in real time on the smartphone (Fig. 2). The additional virtual elements provided more precise directions that the user would follow, this was especially true on the intertwined streets of Sibiu. If you are able to see exactly the direction or path that you need to follow, the possibility of making a bad turn is reduced. We found two major points that we need to work on in order to take Virtual Pathfinder to its next level. First, we need to find a reliable and open access source of maps. Secondly we need to find a way to reduce the battery consumption of the app because using both GPS and the smartphone's camera is quite draining for the battery.

3 Conclusions and further directions of study

The purpose of this research is to study how Augmented Reality can be used to improve the communication of information between computer and users. Our constructive method allowed us to find new features that can be added to a AR app in order to enhanced the interaction with users and to find new directions of study. When we developed Virtual Reader, an AR app based on markers, we discovered that it is possible to compress a lot of information into an AR marker. The marker can be used not only for information storage but also for simulation of virtual elements such as virtual buttons or virtual albums. When we developed Virtual Pathfinder, a markerless AR app, we discovered that AR can be used to improve functionalities that already exist. Furthermore we consider that Virtual Pathfinder provides precise information that couldn't be delivered to the user by any other means than augmented reality. The markerless AR apps need to use additional information to substitute markers. This is the most difficult element of the design of this type of AR apps. One of the most important problem a markerless Ar apps focuses with is the to battery consume problem, due to the necessity of using additional information required to substitute the markers.

In conclusion augmented reality is convenient to be used for providing additional information for the user. It is much more efficient to display information with the help of AR than to input data with AR.

As mentioned by Robert Scoble in his book *The Fourth Transformation* [15] we are approaching a new era.: an era in which we will interact with computers through the use of AR. As a further direction of our study we intent to make a comparison of the AR apps performances taking into account the devices implied in the design and use of these apps.

References

[1] Meagher, R.E. "*STRETCH Report* Computer History " http://archive.computerhistory.org/resources/text/IBM/Stretch/pdfs/05-16/102632634.pdf
[2] Chiappetta, Marco: "AMD Ryzen 7 1800X, 1700X, And 1700 Review And Benchmarks: Zen Brings The Fight Back To Intel".
[3] Words per minute world champion - https://en.wikipedia.org/wiki/Words_per_minute
[4] Neural net headset - https://www.techworld.com/data/what-is-neural-lace-3657074/
[5] Neural Link website - https://www.neuralink.com
[6] Rosenberg, L.B.. "The Use of Virtual Fixtures As Perceptual Overlays to Enhance Operator Performance in Remote Environments". Technical Report AL-TR-0089, USAF Armstrong Laboratory, Wright-Patterson AFB OH, 1992.
[7] Pokemon Go user base - https://www.forbes.com/sites/insertcoin/2017/04/05/believe-it-or-not-pokemon-go-has-65-million-monthly-active-players/#38f36ea9121d
[8] Google Glass - https://en.wikipedia.org/wiki/Google_Glass
[9] Microsoft Hololens - https://www.microsoft.com/en-us/hololens
[10] R. Sandica , P. Bota, , D. Simian - Interactive Reading Experience, The 7th International Conference on Information Science and Information Literacy, Sibiu, Romania, 2016
[11] R. Sandica, P. Bota, Virtual Pathfinder, *Proceedings of the international Conference- "Imagination Creativity Design and Development"*.Sibiu, Romania , 2017. pp. 141-146, 2017
[12]Smartphone ownsership in developed countries -https://www.strategyanalytics.com/access-services/devices/mobile-phones/smartphone/smartphones/market-data/report-detail/global-smartphone-user-penetration-forecast-by-88-countries-2007---2022-UPDATED#.Wm9ZsGluaUl
[13] Unity asset store - https://assetstore.unity.com
[14]Night of Researchers, Sibiu, 2017. https://www.facebook.com/NoapteaCercetatorilorSibiu/
[15] Shel Israel, Robert Scoble (2016). The Fourth Transformation

SĂNDICĂ ROBERT
"Lucian Blaga" University of Sibiu
Faculty of Sciences
Dr. Ratiu Str 5-7
Romania
e-mail: sandica_robert@yahoo.ro

PETRICĂ BOTA
"Lucian Blaga" University of Sibiu
Faculty of Sciences
Dr. Ratiu Str 5-7
Romania
e-mail: botapetre1@gmail.com

DANA SIMIAN
"Lucian Blaga" University of Sibiu
Faculty of Sciences
Research Center ITI
Romania
e-mail: dana.simian@ulbsibiu.ro

Fifth International Conference
Modelling and Development of Intelligent Systems
June 23 - 25, 2017
"Lucian Blaga" University
Sibiu - Romania

An Imperialist Competitive Algorithm Optimized to Solve the Travelling Salesman Problem

Stelian Ciurea

Abstract

We studied the way in which imperialist competitive algorithms (ICAs) can be adapted to solve combinatorial problems. We chose to solve the Travelling Salesman Problem (TSP) (probably the most popular NP-complete problem). In the first stage, we combined ICA with other optimization algorithms (e.g. 2-opt or greedy algorithms) and conducted approximately 13,000 tests to establish values of ICA parameters to ensure optimal behaviour. We then optimized the ICA obtained in the first stage by various methods to increase running speed and improve the results obtained. Finally, we tested the algorithm to solve 11 reference configurations. We consider that the algorithm obtained had a good behaviour: the error obtained for a configuration consisting of over 5,900 cities was 6.54%.

1 The imperialist competitive algorithm to solve the TSP

In September 2007, Esmaeil Atashpaz Gargari and Carlos Lucas presented a paper describing a new type of evolutionary algorithm drawing from history [10]. Called the "Imperialist Competitive Algorithm" (ICA), it has as a model the competition between the imperialist states as well as the way they have led the colonizing policy of political and historical events in the seventeenth, eighteenth and nineteenth centuries. ICA is part of the category of meta-heuristic algorithms based on sets of candidate solutions, also called populations (along with genetic algorithms, cluster algorithms, gravitational search algorithms, etc.).

The standard ICA as presented in [10] is the following:

1. Generate an initial set of countries;
2. Evaluate each country and determine the imperialist countries;
3. Occupy the colonies;
4. Assimilate the colonies;
5. If a colony has better results than the imperialist country then
    a. Interchange the colony with the imperialist country
6. The imperialist competition
    a. Compute the performance of the empires
    b. Occupy the weakest colony of the weakest empire by another empire
    c. If the weakest empire has no colonies left then
7. Remove this empire

8. Revolutions occur in some colonies
9. If the stopping requirements are met then
    a. Stop
10. Otherwise
11. Repeat the algorithm from step 4.

The imperialist country that has the best results following the last iteration is the solution to the problem. Initially, the imperialist competitive algorithm was designed to determine the minimum or maximum of certain functions with one or several real arguments, hence to determine solutions consisting of one or several real numbers. Soon after it was formulated, it started to be applied to solving problems having solutions consisting of whole numbers. In the specialist literature [3], [8], this type of imperialist competitive algorithm was called "discrete algorithm". The problems that fall into this category may be traditional graph theory problems or various practical optimization problems to which solutions with polynomial complexities are not known. Of these, the best-known and the one we chose to solve is the "TSP - Travelling Salesman Problem". We consider it no longer necessary to state it. Theoretically, this is the problem of finding a minimum cost Hamiltonian cycle in a complete graph in which edges have attached costs. An exact solution to this problem can be reached by generating all permutations of the set {1,2,3, ..., n} with 1 fixed point and by calculating the corresponding cost; each permutation is a Hamiltonian cycle. As the number of these permutations is equal to (n-1)!, this method can only be applied for low values of n. TSP is important both theoretically and practically because a number of concrete problems can be formulated as TSP; the most numerous examples can be taken from the integrated circuit manufacturing industry in which situations with higher values for n (n = 744710) [1], [5] have been reported. Various applications can be found in [8]. A pertinent description of the current state of the solutions to this problem, including a list of numerous applications, can be found in [7]. Of the different variants of the TSP, we chose to solve the one called the symmetric Euclidean variant: graph nodes are points in plan, the cost of moving from one node to another is the rounded Euclid distance between the two nodes, and this cost does not depend on the moving direction. The cost of a cycle is the total distance travelled, and this is the function of evaluating a permutation.
The distinguishing features of a discrete ICA occur while implementing the following operations:

1.1 Generating the Initial Set

Each country in the initial set will represent a permutation of the set {1,2,3, ..., n} with 1 fixed point. In the applications that we ran, we chose three values for the initial number of countries: 110, 220 and 550. *We chose a generation based on greedy strategies: the second number of the permutation (corresponding to the second node) was chosen randomly, then at every step we made a list of the x nodes that are closest to the node selected in the previous step and not yet visited (this means that the rounded Euclid distances between these x nodes and the previous selected node are the smallest).* Of these x nodes, the node to be visited in the current step is randomly chosen. This eliminates the disadvantage of poor diversity. The behaviour of the algorithm for $x \in \{1;4\}$ was studied.

1.2 Assimilation

This operation is applied to colony countries. It aims to explore the space of solutions in the neighbourhood of permutations for which the evaluation function has the best values. Through this operation, the parameters that form a colony country are modified so as to converge towards parameters that form a metropolitan country.

In the case of discrete ICAs, in the vast majority of papers that implement this type of algorithm, the assimilation operation is performed as follows: we assume that the solution of the problem that is solved by ICA consists of n natural numbers, each of these numbers belonging to a given finite range. The countries in the algorithm are then combinations of such n natural numbers. Let M be a metropolis of the algorithm and C one of its colonies having the following structure:

M :	m1	m2	…	mp1-1	mp1	…	mp2	mp2+1	…	mn

C :	c1	c2	…	cp1-1	cp1	…	cp2	cp2+1	…	cn

In order to perform the assimilation, two numbers p1 and p2, $1 \leq p1 \leq p2 \leq n$ are randomly generated, and through assimilation, the content of colony C becomes identical to that of metropolis M between positions p1 and p2. The values between positions 1 and p1-1 and p2+1 and n will be those that cannot be found between positions p1 and p2 arranged in the sequence they had in permutation C before the assimilation operation:

C′ :	c1′	c2′	…	cp1-1′	mp1	…	mp2	cp2+1′	…	cn′

$ci' \in \{1,2,...,n\}\setminus\{ mp1, mp1+1 , mp2-1, mp2\}$ and $sign(\sigma(ci)-\sigma(cj)) = sign(\sigma'(ci)-\sigma'(cj))$
where σ and σ′ denote the bijective functions corresponding to permutations C and C′, and $i,j \in \{1,2,...,n\}\setminus\{p1,..,p2\}$.
The complexity of the assimilation operation having this form is Θ(n), provided that the affiliation of a value to set { mp1, mp1+1 , mp2-1, mp2} should be verified in Θ(1) (which is possible if a suitable data structure is used – e.g. a one-dimensional array having the role of a "marker vector": practically, if we note this structure with a, then a[i] = 1 if
$i \in \{ mp1, mp1+1 , mp2-1, mp2\}$, and a[i]=0 if $i \notin \{ mp1, mp1+1 , mp2-1, mp2\}$

1.3 The Revolution Operation in Discrete ICA

We implemented this operation by applying the 2-opt algorithm outlined in [2] in order to rearrange the values of a permutation representing a country. According to this algorithm, for each pair of nodes i and j, $i,j \in \{1,2,...,n\}$ and $i<j$, we test whether, by replacing a cycle corresponding to the permutation of arcs (i,i+1) and (j,j+1) with arcs (i,j) and (i+1,j+1), a smaller length circuit is obtained, and, if so, the circuit actually changes. We should note that nodes i+1,i+2,...,j-1 are covered in the same sequence but in reverse order (see Figure 1). This allows for a quick calculation of the cost of the cycle that would result from this replacement. The structure of a country changes according to this algorithm only if the length of the resulting cycle is smaller. The advantage of applying the method is that very good results are obtained from the very first iterations. The disadvantage is the high probability that the algorithm might lock into a local optimum (usually a value close to the global optimum). The 2-opt algorithm has complexity $\Theta(n^2)$. The whole ICA will have the following complexity:

$$\Theta(no\_iterations\_maximum * no\_countries * n^2) \quad (1)$$

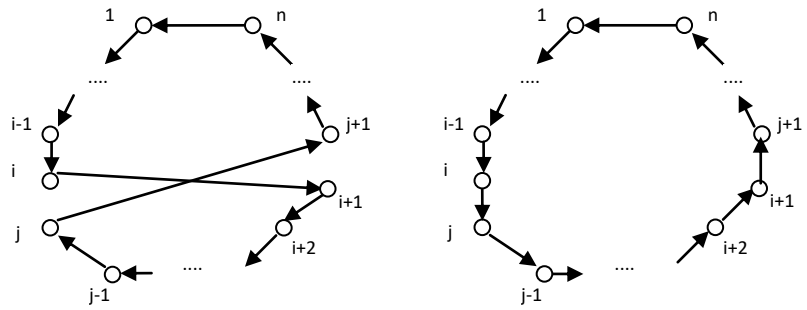


Figure 1-2: opt algorithm

The other operations typical of ICA were implemented according to the formulae presented in numerous articles (e.g. [5]).

2 Sensitivity Analysis on ICA parameters

At the beginning of the tests, we intended to set values for the ICA parameters as well as the implementation variant of the operation of generating the initial set of countries that would determine the optimal behaviour of the algorithm.

The parameters that we aimed to determine were the following: the size of the initial set (nrt), the revolution rate (rr), the approach step used in assimilation operation (p) and the weight with which a colony contributes to the value of an empire's performance (w). Taking into consideration the findings of various papers ([3], [8], [9], [10]), we chose the following sets of values out of which we aimed at selected the value that is the most suitable for these parameters:

- {110, 220, 550} for nrt;
- {4% , 10%} for rr;
- {1 , 3} for p;
- {0.1 , 0.01} for w;

Regarding the method of generating the initial set, we considered four variants based on the algorithm described above in Section 1.1 (x = 1, 2, 3 or 4).

We considered all combinations of these parameters and methods and obtained 96 variants of ICA. For a better appreciation, each of these variants was run for 25 initial sets.
The maximum number of iterations was set at 2000.
We conducted tests for eight configurations, an intrinsic one with 13 nodes and seven reference configurations (downloaded from [4]) with 29, 52, 76, 100, 152, 225 and 442 nodes).
The tests performed indicated the following:
a) The greedy generation algorithm with x=1 provides the best top and average performance;
b) The parameters influencing the performance obtained were the following:

- The revolution rate: a higher value results in better performance;
- The number of countries in the initial set: the performance increases slowly with the increase in the number of countries;
- The extent to which a colony contributes to the performance value of an empire, w, results in better performance if it is low: w = 0.01.

The runtime for the most knotted configuration (442) averaged 9.6 minutes for an initial set of 110 countries, 10.2 minutes for an initial set of 220 countries and 11.2 minutes for an initial set of 550 countries on a computer equipped with an Intel Core i3 microprocessor at 2.93 GHz.

Table 1 shows the results obtained: minL represents the best performance provided by one of the 25 initial sets and avrL represents the average performance of the 25 sets.

configuration	minL	avrL		configuration	minL	avrL
c13	51	51.0		kroA100	21282	21395.3
so29	27603	27603.0		pr152	73682	74135.3
berlin52	7542	7542.0		tsp255	3997	4050.6
pr76	108280	108786.8		d442	52123	52691.9

Table 1: Results obtained by means of the initial variant of ICA

3 Improving the combined algorithm ICA – 2-opt

3.1 Increasing algorithm performance through revolutions

The 2-opt algorithm is very powerful, but it is obvious that if we apply it to the same permutation the second time, its effect will be null. On the other hand, as ICAs run, the potential solutions we work with (the "countries", in the ICA terminology) are "attracted" to some of the best-performing permutations. This will cause the algorithm to stagnate after a fairly small number of iterations. For example, in the 25 tests performed for the 442-node configuration, the average of the last iterations whose lowest cost of a permutation was improved to 636.3.

This observation led us to the idea of improving the performance of the algorithm by modifying the revolution operation: we applied the 2-opt algorithm to 75% of the permutations selected for the revolution, while for the remaining 25% we performed the revolution operation by randomly interchanging some of the elements of the permutations.

The results obtained are presented in Table 2: ICA1 is the initial algorithm and ICA2 is the one with the modification of the implementation of the revolution operation.

configuration	ICA1		ICA2	
	minL	avrL	minL	avrL
so29	27603	27603.0	27603	27603.0
berlin52	7542	7542.0	7542	7542.0
pr76	108280	108786.8	108159	108293.1
kroA100	21282	21395.3	21282	21314.8
pr152	73682	74135.3	73682	73869.6
tsP255	3997	4050.6	3962	3997.7
d442	52123	52691.9	52128	52595.3

Table 2: The effect of the optimization of the revolution

The best behaviour of the ICA2 algorithm for all 8 configurations is evident.

3.2 Reducing the runtime

ICA2 had runtimes of approximately 1.11 hours for 1000-node configurations; a theoretical calculation indicates the value of 111 hours for configurations of 10,000 nodes, which represents a too long runtime. As a result, we aimed to reduce the runtime in various ways:

3.2.1 Software optimization (by means of programming techniques)

This optimization started from the observation made at the beginning of the Section 3 (i.e. if a permutation does not change by applying the 2-opt algorithm, then as long as the permutation does not change by other operations in ICA, there is no point in applying the 2-opt optimization). Hence, we use a one-dimensional array (array) in which we have marked, for each permutation (country), whether or not it was modified or not by applying the operations typical of ICA (assimilation or revolution). In doing so, we avoided the redundant testing of numerous permutations by the 2-opt algorithm. Table 3 shows the average number of permutations (denoted avrnbP) to which the 2-opt procedure was applied in the initial version (ICA2) and in the optimized variant (ICA3), as well as the average runtimes (avrT) in the above-mentioned system for the largest two configurations that we used in the tests performed up to that point. Obviously, the minimum lengths determined by applying the two variants of the algorithm are the same.

	ICA2		ICA3	
configuration	avrnbP	avrT (s)	avrnbP	avrT (s)
tsp225	68816	1257	36322	199
d442	91583	1402	39459	745

Table: 3 The effect of the software optimization of the algorithm

3.2.2 Optimization by increasing convergence speed

The convergence speed of the algorithm can be increased by using a variable weight for each colony that contributes to the performance of the metropolis empire. The variation principle of this parameter is that described in [9]. Thus, the algorithm we proposed for consideration computes the performance of the empires (step 6.a from the algorithm described in Section 1) using the formula:

$$results\_empire = results\_imperialist\_country + \Sigma weight\_colony \times rezult\_colony \quad (2)$$

where the result of a country is the distance travelled following the cycle induced by the permutation (each country represents a permutation).
We assigned to each colony a weight that is initially equal to 0.01. Nevertheless, each time a colony changes imperialist country, the value of the weight of that colony is diminished multiplying it by 0.75.

Table 4 shows the results of ICA with variable weight (ICA4) compared to the results obtained for ICA with constant weight (ICA3). In the tests conducted, the vm1084 configuration with 1084 nodes was also used, downloaded from[4]:

configuration	ICA3			ICA4		
	minL	avrL	avrnbIt	minL	avrL	avrnbIt
pr152	73682	73869.6	2000	73682	73829.3	1269
tsp225	3962	3997.7	2000	3931	3958.5	1429
d442	52128	52595.3	2000	52022	52281.2	1689
vm1084	251125	251403.0	2000	248263	249203.2	1993

Table 4: ICA4 with variable weight vs. ICA3 variable weight

3.2.3 Hardware optimization (by using a super-computer)

Like other evolutionary algorithms based on candidate solution sets, ICA contains an intrinsic parallelism in various stages, for example in the evaluation stage of these candidate solutions. When using a number of np processors, the complexity of the algorithm is as follows:

$$O(nritemax * \frac{nrt}{np} * n^2) \quad (3)$$

The master process and slave processes communicated through the MPI_Send and MPI_Recv functions in the mpi.h library. Table 5 shows the average runtimes obtained with the help of this computer for various configurations as well as the average runtimes by using a computer equipped with a single Intel Core i3 2.93 GHz processor.

	Average runtime (s)		Ratio timePC/timeHPC
configu-ration	PC Intel Core i3	Intel HPC System	
tsp225	195	88.2	2,2
d442	688	137.7	5.0
vm1084	4206	375.1	11.2

Table 5: Average runtime for various configurations

Although the effort to rewrite ICA for a parallel computer is not very substantial, the use of such a computer has an important disadvantage: access is quite problematic. For example, the HPC located at LBU Sibiu was not functional for rather long periods of time.

4 Final results obtained by running an optimized ICA

An ICA having the parameters and optimizations described above was run for different configurations, using other 10 sets of 550 countries. Table 6 shows the results obtained. η represents the efficiency of the algorithm and is an expression that indicates the percentage in which the algorithm improved the minimum length obtained during the first iteration of the algorithm. BKS represents the best known result for that configuration, drawn from [4]; err is the error (in percentage) of the best result compared to the BKS.

configuration	minL	avrL	avrT (s)	η%	BKS	err
so29	27603	27603.0	12.2	5.41	27603	0.00
berlin52	7542	7542.0	17.8	5.42	7542	0.00
pr76	108159	108194.8	67.6	8.34	108159	0.00
kroA100	21282	21297.7	47.5	8.71	21282	0.00
pr152	73682	73758.2	40.3	4.07	73682	0.00
tsp225	3931	3958.5	77.6	12,8	3916	0.38
d442	52022	52281.2	125,0	10,6	50778	3.62
vm1084	248263	249203.2	375,1	10,3	239297	3.74
fl1400	20456	20531.9	548.2	8.64	20127	1.63
d2103	81716	81983.8	761.4	6.88	80450	1.57
rl5915	602564	606257.6	4559.4	6,19	565530	6,54

Figure 1 illustrates how the number of colonies originally held by 50 empires varied in a test performed on a 442-node configuration. We note the convergence of the algorithm that ends after 1796 iterations through the capturing of all colonies by a single empire and the implicit disappearance of the other empires.

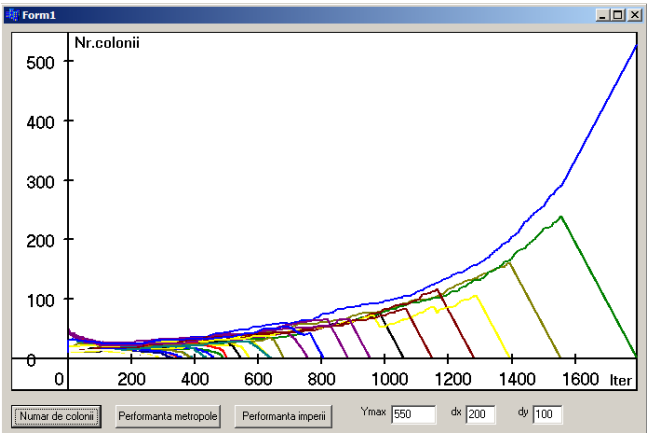


Figure 1: Number of colonies held by the empires of the algorithm dependent on the iteration

Figures 2 and 3 illustrate the evolution of the solution determined by the algorithm for various configurations. The optimum circuits determined during the first iteration, halfway through the number of iterations and at the end, are presented.

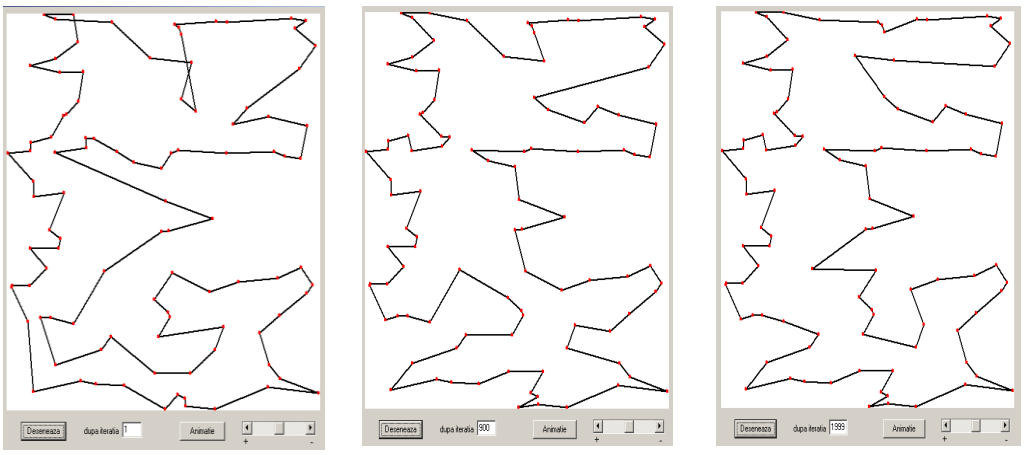


Figure 2: Evolution of the optimal circuit determined by ICA for the kroA100 configuration

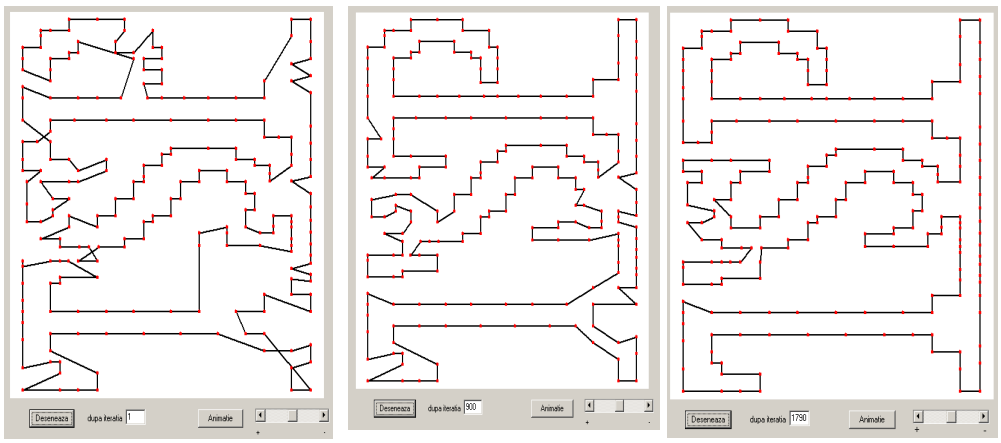


Figure 3: Evolution of the optimal circuit determined by ICA for the tsp225 configuration

5 Conclusions

ICA can be used to solve problems with solutions consisting of a string of natural numbers. The main feature of applying an ICA to this type of problem is that the implementation modalities for some of the characteristic operations differ depending on the actual problem to be solved. Discrete ICA performance can be greatly improved by its combination with other optimization algorithms, such as greedy strategies to generate the initial set of countries, or by implementing specific operations. Through various software and hardware methods, we improved the solutions and reduced the runtimes.

References

[1] Ahmadvand, M., Yousefikhoshbakht, M., Mahmoodi, D. N.: "Solving the Traveling Salesman Problem by an Efficient Hybrid Metaheuristic Algorithm", Journal of Advances in Computer Research, August 2012 , Volume 3, Number 3, pag. 75-84.
[2] Croes, G. A.: "A method for solving traveling salesman problems", Operations Res. 6 (1958) , pag. 791-812.
[3] Hojjat, E., Shahriar, L.: "Graph Colouring Problem Based on Discrete Imperialist Competitive Algorithm", CoRR, 2013.
[4] *** http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/tsp/
[5] *** http://www.math.uwaterloo.ca/tsp/vlsi/
[6] *** http://www.tsp.gatech.edu/apps/index.html
[7] Matai, R., Singh, Surya, Lal Mittal, M.: Traveling Salesman Problem: An Overview of Applications, Formulations and Solution Approaches, http://cdn.intechopen.com/pdfs-wm/12736.pdf (2010).
[8] Shirin, N., Hodais, S., Majid, V. J.: "A Binary Model on the Basis of Imperialist Competitive Algorithm in Order to Solve the Problem of Knapsack 1-0", International Conference on System Engineering and Modeling (ICSEM 2012), IPCSIT vol. 34 (2012), Singapore.
[9] Ciurea S., Trifa V.: "Imperialist Competitive Algorithm with Variable Parameters for the Optimization of a Fuzzy Controller", International Conference on System Theory, Control and Computing ICSTCC, Sinaia, 2014.
[10] Gargari Atashpaz, E., Caro, L.: "Imperialist Competitive Algorithm: An Algorithm for Optimization Inspired by Imperialistic Competition", IEEE Congress on Evolutionary Computation, CEC, 2007.

STELIAN CIUREA
"Lucian Blaga" University of Sibiu
Faculty of Engineering, Department of Computer and Electrical Engineering
E. Cioran Str, No. 4, Sibiu-550025, ROMANIA,
E-mail: stelian.ciurea@ulbsibiu.ro

Fifth International Conference
Modelling and Development of Intelligent Systems
June 23 - 25, 2017
"Lucian Blaga" University
Sibiu - Romania

Why and how Practice impacts Confidence in introductory object oriented programming Courses

Viggo Holmstedt, Shegaw A. Mengiste

Abstract

Why and how do practice of object oriented programming concepts impact self-confidence to instructors teaching introductory object orientation? Does it also impact confidence from students? In this paper, we seek to answer what motivates instructors to build and maintain a codebase over a long time. An aspect that stimulates curiosity is the characteristic of bimodal distribution of marks, a robust characteristic still in search for a conclusive explanation. We do not investigate that characteristic, but refer a few explanations through the study of practicing instructors and their motives. In this investigation, we adopted the notions of social worlds and boundary objects as analytical lenses. We identified two important social worlds: the social world of instructors of introductory object oriented programming and the social world of students introduced to object oriented programming.

Keywords*: Codebase· object oriented programming·self-confidence·credibility·social worlds·boundary objects*

1 Introduction

For many years educational researchers have been attempting to develop models of instruction that specify causal connection between class room teaching behaviours and student achievement [1]. As Ryoo et al [2] pointed out, what challenges educators the most is often not the lack of teaching materials, but the constraints imposed upon a course. These include (1) type of audience, (2) the limited amount of time available for a prearranged set of course objectives, (3) a rigid curriculum, and (4) the choice of teaching method. More specifically, in the context of teaching computer science courses, existing research shows that there have been a number of problems faced by instructors teaching object-oriented programming courses to undergraduate students. Understanding object-oriented concepts like classes, constructor invocation, overloaded constructors, friend functions are difficult for students [3].

A more sensitive aspect of teaching object-oriented programming is the instructors experienced level of abstraction and pace of change in new technologies. Sometimes instructors may not have sufficient experience in the field to teach the subtle and often abstract issues convincingly. It can also be challenging to absorb, adapt and then accommodate new tools adopted in teaching and instruction. This aspect may have sensitive implications, where few instructors would admit that the problem pertains to themselves. If admitted, though, it might influence self-esteem and infer a possible decrease of student confidence and credibility.

The teaching of object oriented introductory programming have intriguing characteristics, like high dropout rates, as reported by Nilsen & Larsen [4]. Another is the mark distribution. Contrary to many other curricula, introductory classes in object oriented programming exhibit bimodal mark distribution [4]. Bimodality here refers to many high and low marks, with few in the middle. The finding of bimodal distribution led to their [4] interest for, and an implementation of, the Keller [5] principle and the associated PSI (Personalized System Instruction). In the original paper, Keller referred to the phenomenon as "upside-down" [5]. Robins [6] reports that the bimodality has been robust through many decades and populations. He further introduced the notion of learning edge momentum (LEM) to explain the factors that contribute to the bimodal distribution of marks. He argues that LEM arises from the interaction of two factors: primarily, successfully acquiring one concept makes learning other closely related concepts easier, but failing to understand one core concept makes further learning harder [ibid 6pp. 40]. Compliance to governmental wishes for a normal distribution of marks (Gauss, single modal, form) is likely to wear down a teacher's self-confidence in terms of setting marks that seldom meet the expectations, not even over long time.

To improve the learning results in introductory object oriented programming courses, some instructors are motivated to develop and maintain a system (a codebase) as a tool to support their teaching. Developing and maintaining this tool type and similar systems presupposes enhanced knowledge and mastery. In this paper, our focus is in exploring how the knowledge and mastery of object oriented programming concepts and the prolonged use of code-based tools affect the self-confidence and credibility of instructors teaching introductory object oriented programming courses. We argue that the characteristics of our research problem may be common to almost all subjects taught on higher levels of education. However, we also think that the bimodal distribution is special to introductory object-oriented programming, because of the subject's characteristics of needing activities with demanding precision and a high abstraction level. We think that few other curricula share this explicit characteristic.

As such, we aim to understand how specific instructor activities, behaviour and actions affect the learning process. More specifically, we want to investigate how the instructors' long engagement and experience in practicing what he/she teaches contribute to build self-esteem and to increase student acceptance and credibility; and to see if this can be augmented by the instructor's long-term activities of local codebase maintenance. We do not measure the learning results, nor the mark distribution, but claim by abduction that the self-esteem and student confidence and credibility will benefit the teaching activity. We investigated the problem based on data gathered from 10 instructors employed at several higher learning institutions in Norway, who had many years of experience in teaching object-oriented programming. Some of our respondents adopted codebase to support their teaching activities. Besides, we have also used the many years experiences of one of the authors of this paper who have been engaged in creating and using a codebase called Evalanche [1][7].

As analytical framework, we used concepts of social worlds and boundary objects from the literature of symbolic interactionism [8-12]. The social worlds perspective is a well-established and rich sociological tradition to investigate and analyze teaching and learning as a socio-technical process [13, 14] that involves participation of different actors.

The remaining structure of the paper is as follows: Section 2 provides an overview on our conceptual framework. Section 3 introduces the research approach and methods employed. Section 4 presents the empirical findings. The last section, presents the analysis, discussion, and implications of the study.

[1] Evalanche: an LMS supporting teaching and enhancing learning as well as improving self-confidence and credibility among students.

2 Conceptual Framework: Social Worlds & Boundary Objects

The concept of 'social worlds' has its roots in the American sociological tradition of symbolic interactionism [11, 15, 16] which primarily focuses on meaning construction "amongst groups of actors – collectives of various sorts – and on collective action — people doing things together and working with shared objects" [9]. The emphasis on the creation of meanings through interaction has strong parallels with the social construction of technology (SCOT) [17]. The key argument in both symbolic interactionism and social construction of technology is that the development of technological artifacts is an outcome of complex social and political negotiations [13]. Another important concept introduced by the social world's perspective is the notion of arena. An arena is a field of action and interaction among a potentially wide variety of collective actors (be it human, organizational or technical artifacts). In arena, the different social worlds that focus on a given issue and prepared to act in some way are brought together [11, 18]. Gieryn [19] also identified three important properties that are common to all social worlds: the potential for division and segmentation into sub-worlds; intersection with other social worlds; and legitimization through the definition and enforcement of standards and boundaries of a social world.

The concept of boundary objects appears to investigate and analyze boundary-spanning phenomena in organizations. Boundary spanning describes activities that occur at the *boundaries* between different groups with different tasks and activities and with a need to cooperate for a common goal [20]. When there are two or more groups with boundary spanning activities (tasks), establishing meaningful communication and collaboration between those groups inevitably poses significant challenges. There are usually challenges associated with transferring [21, 22] and translating information from one world to another; there are problems of information integrity and translation.

Star and Griesemer [23] argued that certain boundary objects are flexible enough to accommodate different interpretations or interests emanating from various distinct social worlds, yet robust and plastic enough to maintain a common identity across all social contexts, then allowing translation to take place across the boundary. Gal, Yoo, and Boland [24, 25] describe boundary objects as:

> *"… abstract or physical artefacts which reside in the interfaces between organizations or social communities and have the capacity to bridge perceptual and practical differences among diverse communities in order to reach common understandings and effective cooperation" (Gal et al., 2004, pp. 194).*

The notion of boundary objects have been widely used to explore a broad range of settings where boundary-spanning activities are critical. For example, Henderson [26] used engineering drawings and sketches as boundary objects to enable communication across different occupational communities. Pawlowski & Robey [27] used the notion of boundary objects to analyze the knowledge brokering work of IT professionals. Tudhope et al [28] also illustrate prototypes as important boundary objects in software development activities, since they facilitate negotiation and communication practices between users and developers. Various artefacts and representations in software development practices mediate the communication between different groups and the process of synthesizing their diverse interests; e.g. between designers, between designers and managers, between designers and end-users [20]. In those contexts, boundary objects need to provide the basis or common ground, which different groups share and use to shift their "knowledge boundaries" in order to solve their joint heterogeneous problems (ibid, 2006).

In the context of the study, we adopted the notions of social worlds and boundary objects as our analytical lenses. We identified two important social worlds: the social world of instructors and the social world of students. The two social worlds intersect where they are both concerned about the learned outcome. The social world of instructors introduces and executes the novelty aspect [29], while the student world is challenged by the same novelty. The challenge of the novelty lies in the fact that programming concepts become more abstract along the learning path, and that the rules for best

practices become harder to accept, and even believe, for novice programmers. The concerns are divergent in the sense that the instructors world want to teach methodically and systematically, while the student world want to execute learning steps fast, as in finishing a list of exercises. In this divergence of concerns, there is a great need for confidence and credibility to impose and accept each other's aims. A boundary object is therefore one that ensures self-confidence in the instructor, as well as confidence and credibility in the student towards the instructor.

3 Research Approach and Methods

3.1 Research approach

This study employed an engaged scholarship approach [30], which seeks to combine both qualitative and quantitative methods. We believe that this approach offers us both the methodological and analytical leverage in our investigation.

Van de Ven defines engaged scholarship as "a participative form of research for obtaining the different perspectives of key stakeholders (researchers, users, clients, sponsors, and practitioners) in studying complex problems" [ibid30]. This approach, according to Van de Ven [30] aims to move beyond the simplistic notion that research knowledge is generated in the lab and then packaged and diffused into practice. Rather, Van de Ven adopts an interactional view in which professional and research practices contribute to each other's growth through different types of activity. By embracing qualitative as well as quantitative methods and promoting process studies as well as variance studies, the notion of engaged scholarship offers opportunities to transcend the traditional dichotomies of core versus diversity, rigor versus relevance, and positivist versus interpretive. Moreover, Van de Ven [30] identified four different forms of engaged scholarship: *informed basic research, collaborative basic research, design and evaluation research, and action research (ibid).* In this research, we categorized our research as an informed basic research that aims to describe specific aspects of the social phenomena teaching and learning.

3.2 Data Collection Methods

The study has relied on two prominent sources for empirical material. Primarily we interviewed 10 instructors from higher learning institutions in Norway who have been teaching introductory object-oriented courses. The data collection was in Norwegian, and later translated to English. Our secondary source is the experiences of the first author of this paper. From 1999 to 2017 he developed the codebase Evalanche to support his teaching. His experiences contributed a lot in giving interesting insights and findings for the scope and purpose of this paper.

Informal Meetings with Instructors

Informal discussion with colleagues who have been engaged in teaching introductory object-oriented courses triggered this study. As most of the colleagues had many years of teaching as well as relevant practice in teaching object-oriented programming courses, it became interesting to approach some colleagues to discuss their experiences. At first, the discussions took place in informal meetings from 2009 to 2012. The discourse revealed that some of those colleagues practiced object oriented programming, as well as built software to either administrate their teaching, or to support some other aspect of the student's needs. Most of those colleagues claimed that the overall teaching and learning outcomes would likely benefit from it. Then the idea matured to design a research agenda targeting the teaching of introductory object-oriented programming in combination with maintaining code relevant to the teaching and learning activities.

For further exploration of the topic and the problems involved, we conducted telephone interviews with 12 instructors. During an informal introductory interview, we discussed a respondent's relevance of practice to find out whether it would suit the investigation. We then contacted the 12 instructors through email to let each decide on a further participation in the investigation. This approach left us with 10 relevant respondents.

Questionnaire

We prepared a questionnaire with a guide and deployed to the 12 candidate instructors. Before answering the questionnaire, the respondents considered three premises:

- Did the coding activities have a duration equivalence of more than 12 months practice?
- Were the coding activities relevant to teaching undergraduate object oriented programming, administrative activities included?
- Has the coding activities influenced a perception of vocational self-confidence, ability to teach and general role relevant to teaching, including credibility and confidence from students?

Two candidates exited the investigation, while 10 respondents remained. The 10 remaining candidates qualified to answer the questionnaire.

Longitudinal experience

As our second source of data, we also relied on the data collected from the many years of experiences of the first author of this paper in both teaching undergraduate object-oriented programming courses and in developing the codebase Evalanche [7] to support the teaching activities. The values of the experience pertain to observations in several domains, like technology, acceptance from the relevant social worlds, compliance to social rules, user demands on reliable services.

3.3 Data Analysis Technique

Established methods for handling qualitative data inspired our analysis of the empirical data. NVivo, using coding to nodes as a main approach, supported the responses analysis. The coding helped us localize relevant statements, sometimes related across the actual questions. We read carefully through the interviews from the questionnaire to get an insight of the total material. To inspect concrete themes related to our research question, we coded fragments from the responses, and let NVivo help us find relationships and contexts across respondents and questions. The analytical framework, which embraced the notions of social worlds and boundary objects, has been used to inform the NVivo analysis and thereby served as a sensitizing device for what 'to look for' in the data. Based on those analysis techniques, we were able to find relevant statements to present in the findings section.

4 Findings

In this section, we present our findings from the interviews and the personal experiences of one of the authors. Our empirical evidence has two prominent sources. The first source is the questionnaire, with evidence from instructors presented in section 4.1.

The second source is from the first author's experiences of gaining practice, insight and knowledge related to development, maintenance, using and administering of a learning and teaching management system through 18 years, presented in section 4.2.

4.1 First source: Responses from Instructors

System types

From the coding of answers to nodes, we could find out more about what kind of systems the respondents had worked with.

a. A course in study technique as part of the Keller based experiment [4]

b. An ambitious tool for collecting several learning resources [31]. The tool collects, organizes and then presents diverse instructional material from sources of many different formats

c. Administrative tools for salary negotiations and tools for individual work plans

d. A variety of tools for research or professional activities

e. An ambitious tool for the administration of learning and teaching [7]

With these types of systems in mind, we will study relevant evidence from the responses repository.

Effects of creating and maintenance of codebase

Most respondents reported they have gained from the coding activities in terms of both new skills and confidence. A few respondents link their general level of profession to their high level of skills and competence. More typically, one respondent states: *"Doing big development projects is by its nature positive, especially since the development is within technologies I am teaching daily."* Statements like "*Developing big systems means much coding and work that we tell the students about*" sustain the development of personal skills and coding competence.

Most respondents have gained relevant technology experience from the very coding activities. Statements like "*One will meet problems [...] to overcome. By this both skills and competence grows.*" posed in the learning arena, generate a feeling of credibility from students. One statement is "*This [the credibility] is connected to my own professional self-esteem. But also the fact that one can refer to having developed big systems lets one get a natural authority.*" In general, the long work with systems feels important: "*I have experienced that my students have respect for my long practice*". Several respondents report a positive effect of better and more interesting teaching methods due to their practice of maintaining code to complement their teaching.

Most respondents clearly relate augmented confidence to their coding and maintenance activities, as in the statement: *"maintaining the code is important as it gives me hands-on experience with new methods, new tools and new document formats."* This is also important for credibility, further supported by the next evidence: *" I can confidently say [to the students] that I have experience with ... And I can elaborate more".*

Attitudes of respondents towards their role

We learned about the respondents' perception of their role and social mission linked to maintaining code. An important statement is this: "*I mean that the project could be useful to quite many, and that it could contribute to considerable cooperation and sharing of experience.*" There is also an interesting dimension of duty. "*I should be able to develop tools that I need.*" Statements like "*I think it is my duty to be updated within my field*" demonstrate a sense of commitment to the role apart from personal concerns.

Some also indicated their concerns for still more: "*I think it is important, not least the overarching aspects*". Through a personal follow-up of this statement, we learned that the expression "overarching aspects" referred to concerns for the quality of educated system developers. We also found benefits to the educational economics: "*I believe the system has caused an increase in the production of credits*". One of the respondents gave a quite challenging piece of evidence: "*I think that if possible, all programming instructors of higher level should practice* [programming skills] *in the industry from time to time, or at least take an active part in programming projects, as for instance Open Source projects*."

Motivation for developing and maintaining codebase

All respondents had personal interest in mastering several technologies. Still it is interesting to learn that no respondent let this be the sole motive. "*No [it was not my own needs], but it was a good bonus.)*". For the maintenance of administrative tools, we find the interesting statement: "*... I have selected methods and tools relevant to me in my teaching, as well as relevant to the development in the subject I teach.*"

We also found very friendly statements concerning the amount of *quality material made accessible* by one of the tools: "*Those examples and explanations are accessible for the students and to myself when constructing lessons and demonstrations*". The responses indicate that coding instructors generally think their efforts have positive influence. One respondent expressed that "*My own need for doing something good for the students is a rational motive [for maintaining of the code] for me*".

4.2 Second source: Gained insight from development

Developing a boundary Object (the Evalanche System)

The first author of the paper began coding a tool for the administration of learning and teaching, Evalanche [7], in 1999 to gain better teaching skills in Java programming. The ambitions to develop the system were also:

a. to give undergraduate students in college a fast feedback on a great number of exercises;
b. to conquer administrative friction related to the publishing of exercises, collecting answers and computing the results by creating the shortest possible path to the student's work;
c. to support iterative and incremental guidance, where students could involve themselves in a dialog tied to each exercise and each try. The file-set submitted by a student for each try, should also be persisted and easily retrieved anytime.

As boundary object, Evalanche has helped to facilitate skill and knowledge transfer from instructor to student. Since the maintenance was just as much a coding training arena as a tool to support coaching of students, several technologies were exploited to satisfy current needs. The boundary object convey informational objects concerning both social worlds. The handling of students goes on continually. The instructor publishes exercises, the student submits solutions, and the instructor delivers individual guidance and assessment in repeating cycles for each student, each exercise and each try for as long as the student needs. The boundary object emits friendly emails to remind procrastinators. The coder continually extends the boundary object in several interesting ways. A recent example is using IoT, realized by signalling a smart wrist device, to give the instructor an early submission alarm in order to deliver the quickest feedback possible to the student.

Effects of creating and maintaining of codebase

Some instructors of vocational skills may be subject to incredibility caused by the instructor's distance to practical work. Credibility may improve, though, if the instructor can refer to earlier, preferably *recent* practice relevant for the taught topics. The Evalanche codebase maintenance inferred multiple ways and cases to boost the interest and involvement in many existing technical fields, as well as motivation to be exposed to upcoming technologies.

Pattern archaeology, detection and accommodation have been huge resources for tailoring the code. When stuck somewhere in the code, implementing a better pattern for the mechanism in question has often solved the problem. Even *considering* another pattern may be enough to unleash better semantics to help solve a complex problem. Many times bad semantics lead to a state of misunderstanding. Maintaining code infers an appreciation of good semantics. This piece of experience and several more have in turn contributed to the development of specific courses and instruction material in *Software Design Patterns for Object Oriented Systems* [32, 33].

The numerous repeating cycles of counselling for each student, each exercise and each try can improve the student's power of abstract thinking. The boundary object's ability to let the instructor drill students in specific problems improves generalisation and abstract thinking. The iterating and intense training over time add the abstraction ability to the student's problem solving characteristics. The students' ability of abstraction may contribute to reducing marks bimodality [4, 6].

Object oriented problem solving require practical skills as well as a degree of abstraction sometimes scarce among young students. Kunkle [34] states that the choice of a specific introductory language or tool has small effect on the learning results. There is a possibility that a large amount of repetitions stimulates the abstraction abilities needed to understand the solution. The boundary object does exactly that.

A main ambition from the very beginning of the project in 1999 was to remove most of the administrative friction between *emitting exercises* and *finishing their evaluation*. This kind of friction is described in more detail in [7], where there is also a brief discussion of further advantages of

reduced administrative friction. The boundary object allows intense training by quickly connecting instructor, student and submission. Exploiting this characteristic enhances the student learning as well as the credibility and even enthusiasm in the student social world.

Attitudes towards role

A motivation inferred by the teacher role was to let Evalanche boundary object open the shortest possible path between submission and assessment. Another motivation was to engage both social worlds in building a better system, and discussing coding issues affecting their own user experience and specific technical curiosity. The system has registered several instructors. But few have used it over time, similarly to what is stated for System Type b [31], section 4.1. This experience has contributed to knowledge of how to gain peer acceptance for in-house software. Real world experiences from maintaining and *running* multi-user software, have improved the role as instructor and author, as well as helped develop robust attitudes to the social mission aspects of system development.

Among strategies for augmenting acceptance for boundary objects like Evalanche, we suggest to involve one or more applications of artificial intelligence. According to Gartner's Hype Cycle from July 2017, the concept of AI is still ascending towards the top, while ML is just over the top, still 2-5 years before disillusionment. In this perspective, an optimization of the boundary object in terms of AI services would be a mainstream strategy applied at the right time. We elaborate this strategy in section 7, Future research

5 Analysis & Discussion

We have investigated why and how long practice impacts confidence in introductory object oriented programming courses. The instructors under investigation had created boundary objects that inferred self-esteem and perceived student confidence in several ways, positively stimulating learning processes in both social worlds. We kept the intersection between the social worlds of instructors and students of introductory courses of object oriented programming stable, while the boundary objects varied. The 10 respondents had created diverse boundary objects with the common characteristic of augmenting self-confidence and professional experience onto the instructor world, and perceived confidence from the student world. The boundary objects had these characteristics for a variety of reasons. Some added facilities that satisfied the informational requirements or needs of each of the social worlds. Others invoked professional respect by virtue of their size and complexity. We will now enhance the table in section 4 to analyse the system types, and to facilitate a discussion of how specific system types satisfy needs of the social worlds.

Type	
a	This boundary object type is strongly structured within the experiment at UiA [4]. The boundary object was the foundation of a pedagogical experiment targeting bimodal results in introductory programming courses. The report [ibid 4] points out a few weaknesses in the use of the object, and lets us know that the "new course format was a moderate success".
b	This ambitious boundary object concentrates on conveying and assembling learning materials from any thinkable open source. As such that object fits the success criteria of satisfying informational requirements, as claimed by Adam Worrall [35] in his analysis of Star and Griesemer [23] and Meleis [36]. In addition, the construction of the object demanded by its nature a variety of specialized technical insights and colleague cooperation, and thus the boundary object helped transferring the learning results among both colleagues and students.
c	This boundary object type creates distance to the transfer of knowledge between the social worlds of interest here. Instead, the objects engages almost all colleagues from all faculties within one campus, since they affect important parts of the local administration. Therefore,

	their construction infer general professional confidence and a lot of diverse technical experience.
d	For the researcher members of our instructor world, the boundary objects are research tools. They usually satisfy informational requirements either directly or through research reports. The many interesting and very specialized informational needs in that field infers professional respect on restricted and high level arenas.
e	As discussed in section 4.2, a main idea was to support *fast* response to quite *many* submits, and let the student improve the answers iteratively. This may influence what learning edge [1] each student is drawn towards at an early stage.

Characteristics that influence positive perceptions from both social worlds are duration and pedagogical relevance of boundary objects. The commitment to boundary object maintenance is likely to be motivated by the common targets of the two social worlds. The engaged instructor is motivated by the learning outcome, as well as the perception of confidence from the student social world. This assertion strongly conforms to the analysed evidence. Tin et. al [37] found that more than half of the teachers were motivated by students and by teaching itself. A basic assertion is that motivation is good for teaching. We will also assert the positive influence of ownership to your job as a factor that regulates motivation. Hutabarat [38] discusses motivation and performance as important for organization success and outcomes, and states that there is a significant direct effect of job-satisfaction to job-performance. Boundary object maintenance belongs to the factors that enhance the notion of ownership, and thus influences the notion of ownership. The possibility of spending resources on boundary objects rests on confidence and autonomy at work. Teachers in Hutbarat et al. were motivated because principals trusted them and gave them considerable autonomy.

The creation and management of boundary objects is a key process in developing and maintaining coherence across intersecting social worlds [23]. In the intersection of our social worlds, there are some surprisingly difficult knowledge transfer issues. At first, there are syntactic challenges. The transfer of knowledge must happen slowly enough for the student world to accept and incorporate it. After some kairos [39] time shifts, the boundary object shifts state to semantic, and then to pragmatic [29] state. All this happens along the increase in abstraction of the learning challenges. All along the increasing novelty charge [ibid 29], the boundary object must execute its capacity to spawn self-confidence and credibility power and effect within the social worlds. For external codebases this is done through already accumulated reserves of experience, like a few years of industrial experience. For internal codebases it is more likely to be a continuous process with renewable energy and technology as demands to the running systems development activities.

We think our investigation indicates that weakly structured boundary objects of the types we have related, work well in the transfer of knowledge between the two social worlds. In most of their strong structured forms in the intersection of the social worlds, they infer self-confidence and a feeling of student credibility and confidence. From this conclusion, we think that instructors should be encouraged, or at least not discouraged, if they practice relevant coding. In hind sight, we are also convinced that our choice of the social worlds perspective gave us the opportunity to examine the experiences and learning processes that occur in shifting and open constellations of actors. This opens up the broader context as well as the temporal dynamics for analysis [9, 18, 40].

Our analysis drew on the notion of boundary object; a notion that has enjoyed a substantial take-up outside STS [41]. In the original formulation of the notion, the interpretive flexibility was a core characteristic of a boundary object. This would facilitate ‘cooperation without consensus’, i.e. communication and/or coordination could be achieved between different social worlds. However, the worlds would remain different and autonomous. The notion of boundary objects could explain exactly this balance of autonomy with collective action. It is also worthwhile to note that these boundary objects were seen to be the outcomes (rather than causes) of processes where convergence between the

needs of the different worlds had been achieved and maintained [42]. Other studies, such as [29, 43] also recognize boundary objects as 'causes', as instruments to achieve collaboration, and also as means to facilitate knowledge transfer and learning. Our findings also revealed that boundary objects like Evalanche serve as outcomes of the code development process and set the ground to establish collaboration between the student and instructor worlds. It was also an instrument to facilitate knowledge transfer and build self-esteem and confidence for instructors. In our case, the boundary object facilitated learning not merely through mediating or aligning the worlds of students and instructors. The meeting arenas that emerged around the codebase did not just facilitate a 'transfer' of knowledge between different worlds. A more central role was that these meetings between social worlds generated situations where different meanings were exposed. The multiple ways of presentation, examining and processing of concrete boundary objects, let the differences of values, opinions, frames, expectations and preferences become visible. A major role of the boundary object was thus to uncover tensions and to stimulate collaboration. This goes beyond mere mediation [44].

6 Threats to validity

The evidence for this qualitative research is from carefully selected and scarce respondents, as well as one author's reported experiences. In Onwuegbize and Leech [45] we can investigate validity in terms of qualitative research. We learn that interpretivists are encouraged to improve upon frameworks like the Qualitative Legitimation Model [46]. However, our evidence stems from a population being very small by its nature. We therefore claim the results to be interpreted as findings from an exotic and interesting department of the social worlds under investigation. We found strong evidence of positive influence on self-esteem and perceived confidence. But, is it even possible to find negative perceptions of confidence when the respondents are already selected by their confirmation of a perception of vocational self-confidence? One respondent who claimed his self-confidence to originate from his professional grade and situation, rather than from the creation of boundary objects, opens for exactly that possibility. The existence of confirmation bias still threatens this interpretation, since the data set violates a reasonable compliance to rich and thick descriptions, as discussed as early as 1970 by Becker [47].

Onwuegbize and Leech encourages "interpretivists to document how they obtained their data, their interpretations, and their conclusions." The degree to which we have failed to document the data collection and interpretation therefore influences a possible threat to validity.

7 Future research

Our research ambition is to invoke further research and discussions of effects of long term code maintenance for instructors of introductory programming courses, as well as to establish an opinion on the usefulness of such activities. We would like to have repeated the research presented here at a bigger scale, involving a much broader international respondent universe than we could find in Norway at this time.

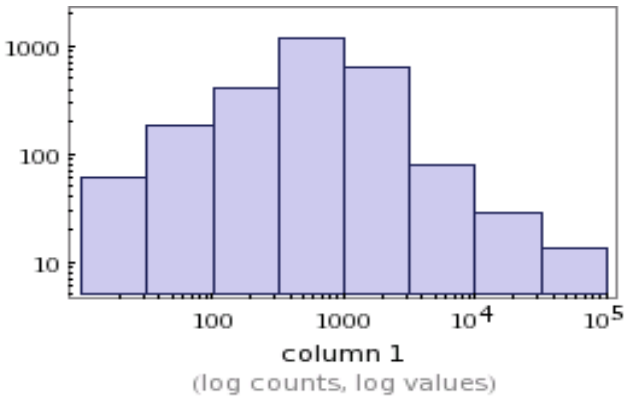
We would also like to investigate the LEM suggested by Robins [1], and link experiments with very fast responses to student submissions. If it is true that the effect of Learning Edge Momentum is particularly strong in introductory programming, then it is of interest to find out if early response could attract students towards the positive edge.

Tools like Evalanche support iterative guidance and assessment to help individual students achieve high levels of marks and enhanced understanding of the exercise. We would like to investigate whether iterative guidance is sustainable in institutional and societal perspectives, since iterative guidance and assessment demand considerable resources from the instructor. The instructor would benefit from predicting exactly which students will benefit from the resources in terms of completing mandatory work, in cases where mandatory work is a prerequisite for admittance to the final exams.

Instructors could then spend more resources on students with the better possibilities. The present situation is that instructors sometimes spend resources randomly and in vain.

Evalanche measures several parameters, as for instance the lag for the student first response on an exercise, as well as the lag for the student's first response to his submission from the instructor. Facilitating short lags is an explicit attribute of Evalanche, for the purpose of enhancing the student learning, motivation and the promotion of positive LEM effects.

To give an idea of response intervals over a long period we created a simple diagram showing the amount of first responses vertically against delay in minutes horizontally. The total amount of first responses is 4060 for the period 2006 to 2011.



The mean of responses is 1599, where the response delay is around 1000 minutes. Such a response frequency combined with low delays is made possible by reducing administrative friction[7]. Still, an interesting question is whether the high response rate is worth the effort. Following this question, it becomes even more interesting to consider ML techniques as an application of AI to provide predictions of the success rate in terms of student mandatory work.

We indicate a simple design of supervised learning of the outcome of response rates in the simple table below. The actual implementation is slightly more involved.

Student	diff(T(submit),T(served)) in minutes	Mandatory Success
1	10	true
2	200	false
3	320	true
4	1020	true
5	23000	false
6	8	false

We indicate with this table that both high lag values and very low lag values for an exercise may combine with false values. Even considering that each student will have measures from several exercises, we would have to decide on using the average experience of delay, the summed total, or put

more weight on the position in time for the measured lags. Anyway, we would hesitate on predicting a settled failure based on this design alone.

A more robust approach would be to access and engage all available student behavioural measures in Evalanche, to build a neural network of 0..1 values for perceptrons, with the outcome of predicitive analytics for mandatory success. In a multi-layered neuron design we would consider adjustable eights, identified by weight limits to predict whether a student with such output is likely to be admitted to the final exams. This design involves learning algorithms on the several layers of neurons. After a learning period supplied with tests we would decide whether it was actually possible to predict the chance of mandatory success based on social world behaviour for this boundary object. If not, it would be open to further research to optimize the boundary object for more precise and targeted measures of student behaviour in order to balance resource usage on individuals in the student social world.

Comparable open source LMS instances

Several interesting open source LMS instances exist as comparable boundary objects among similar social worlds as ours. A few examples of standardized tools are Moodle, ILIAS and Sakai, and less open source projects like Khan Academy and Canvas. They all go far beyond Evalanche in marketing, purpose and appeal. The first coding of ILIAS happened almost simultaneously with Evalanche. ILIAS reached SCORM [48] compliance a few years later. Moodle was released around 2002, and Sakai a little later. Those LMS structure learning contents well. Evalanche do not, but emphasizes fast and personalized instructor feedback.

An instructor from our social world could benefit from participating among the open source developers of one of the standardized LMS tools. The perceived effects of creating and maintaining codebase fragments could even be better, or at least similar, as member of a developer community, compared to working inside the more limited and private Evalanche project. The actual chosen LMS and its policies would, however, limit the freedom to explore several very different technologies, communication variances and presentation forms. There would also still be an issue to have colleagues, and the institution, to accept same LMS.

References:

[1] Robins, A., *Learning edge momentum: a new account of outcomes in CS1.* Computer Science Education, 2010. **20**(1): p. 37-71.

[2] Ryoo, J., F. Fonseca, and D.S. Janzen. *Teaching Object-Oriented Software Engineering through Problem-Based Learning in the Context of Game Design*. in *Software Engineering Education and Training, 2008. CSEET '08. IEEE 21st Conference on*. 2008.

[3] Holland, S., R. Griffiths, and M. Woodman, *Avoiding object misconceptions*, in *Proceedings of the twenty-eighth SIGCSE technical symposium on Computer science education*. 1997, ACM: San Jose, California, USA. p. 131-134.

[4] Nilsen, H. and E.Å. Larsen, *Using the Personalized System of Instruction in an Introductory Programming Course.* NOKOBIT, 2011. **2011**.

[5] Keller, F.S., *Good-bye, teacher.* Journal of Applied behavior analysis, 1968. **1**(1): p. 79.

[6] Robins, A., *Learning Edge Momentum*, in *Encyclopedia of the Sciences of Learning*, N.M. Seel, Editor. 2012, Springer US: Boston, MA. p. 1845-1848.

[7] Holmstedt, V., *Regulering av oppgavefrekvens gjennom automatisering*. Vol. 4/2002. 2002, Tønsberg: Høgskolen i Vestfold.

[8] Clarke, A.E., *AS ORGANIZATIONAL THEORY.* Social organization and social process: Essays in honor of Anselm Strauss, 1991. **119**.

[9] Clarke, A.E. and S.L. Star, *The social worlds framework: A theory/methods package.* The Handbook of Science & Technology Studies, 2008. **3**: p. 113-137.

[10] Mark, G. and S. Poltrock, *Groupware adoption in a distributed organization: transporting and transforming technology through social worlds.* Information and Organization, 2004. **14**(4): p. 297-327.

[11] Strauss, A.L., *A Social Worlds Perspective.* Studies in Symbolic Interaction, 1978. **1**: p. 119-128.

[12] Strauss, A.L., *Social Worlds and Legitimization Processes*. Studies in Symbolic Interaction, ed. N. Denzin. 1982, Greenwich CT: JAI Press.

[13] Garrety, K. and R. Badham, *The Politics of Socio-technical Intervention: An Interactionist View.* Technology Analysis & Strategic Management, 2000. **12**(1): p. 103-118.
[14] Strauss, A.L., *Continual permutations of action Aldine de Gruyter.* New York, 1993.
[15] Shibutani, T., *Reference Groups as Perspectives.* American Journal of Sociology, 1955. **60**(6): p. 562-569.
[16] Unruh, D.R., *The Nature of Social Worlds.* Sociological Perspectives, 1980. **23**(3): p. 271-296.
[17] Bijker, W.E., T.P. Hughes, and T.J. Pinch, *The Social Construction of Technological Systems: New Directions in the Sociology and History of Technology*. 1987: MIT Press.
[18] Strauss, A.L., *Scientists and the evolution of policy arenas: The case of AIDS.*, in *stone symposium of the society for the study of symbolic interaction*. 1991: San Fransico, CA.
[19] Gieryn, T.F., *Boundaries of science.* Handbook of science and technology studies, 1995. **393**.
[20] Gkeredakis, E. and K. Samiotis. *The Process of Creating Boundary Objects: The Case of a Knowledge Management IT Artefact*. in *OLKC 2006 Conference at the University of Warwick, Coventry on 20th–22nd March*. 2006. Citeseer.
[21] Von Hippel, E., *"Sticky information" and the locus of problem solving: implications for innovation.* Management science, 1994. **40**(4): p. 429-439.
[22] Szulanski, G., *Exploring internal stickiness: Impediments to the transfer of best practice within the firm.* Strategic management journal, 1996. **17**(S2): p. 27-43.
[23] Star, S.L. and J.R. Griesemer, *Institutional ecology,translations' and boundary objects: Amateurs and professionals in Berkeley's Museum of Vertebrate Zoology, 1907-39.* Social studies of science, 1989. **19**(3): p. 387-420.
[24] Gal, U., Y. Yoo, and R. Boland Jr, *The Dynamics of Boundary Objects, Social Infastructures and Social Identities*, in *Sprouts Working papers on information systems,*. 2004, ase Western Reserve University: Cleveland, Ohio.
[25] Gal, U., Y. Yoo, and R. Boland Jr, *The Dynamics of Boundary Objects, Social Infastructures and Social Identities.* ECIS 2005 Proceedings, 2005: p. 57.
[26] Henderson, K., *Flexible sketches and inflexible data bases: Visual communication, conscription devices, and boundary objects in design engineering.* Science, technology & human values, 1991. **16**(4): p. 448-473.
[27] Pawlowski, S.D. and D. Robey, *Bridging user organizations: Knowledge brokering and the work of information technology professionals.* MIS quarterly, 2004: p. 645-672.
[28] Tudhope, D., P. Beynon-Davies, and H. Mackay, *Prototyping praxis: Constructing computer systems and building belief.* Human-Computer Interaction, 2000. **15**(4): p. 353-383.
[29] Carlile, P.R., *Transferring, translating, and transforming: An integrative framework for managing knowledge across boundaries.* Organization science, 2004. **15**(5): p. 555-568.
[30] Van de Ven, A.H., *Engaged scholarship : a guide for organizational and social research*. 2007, Oxford ; New York: Oxford University Press. xii, 330 p.
[31] Stenseth, B. *WXT*. 2009 2009 [cited 2016 2016-03-23]; Publishing resource]. Available from: http://www.ia.hiof.no/wxt/wxtsite/.
[32] Holmstedt, V., *Mønstre og tekniske applikasjoner*. 2007, Bergen: Fagbokforlaget, Norway.
[33] Holmstedt, V. *Objectification*. 2010 [cited 2017 22 March]; List of teaching materials]. Available from: https://sites.google.com/site/mitsoptew/.
[34] Kunkle, W.M., *The impact of different teaching approaches and languages on student learning of introductory programming concepts*. 2010: Drexel University.
[35] Worrall, A., *Boundary Object Theory: Concepts, Propositions, and Limitations* Florida University: Florida University. p. 20.
[36] Meleis, A.I., *A model for evaluation of theories: Description, analysis, and critique of theory.* , in *Theoretical nursing: Development and progress* J.B. Lippincott, Editor. 1991, J. B. Lippincott: Philadelphia. p. 213-245.
[37] Tin, L.G., L.L. Hean, and Y.L. Leng, *What motivates teachers.* New Horizons in Education, 1996. **37**(1): p. 19-27.
[38] Hutabarat, W., *Investigation of teacher job-performance model: Organizational culture, work motivation and job-satisfaction.* Asian Social Science, 2015. **11**(18): p. 295.
[39] Jaques, E., *The form of time*. 1982: Crane, Russak New York.
[40] Clarke, A., *Situational analysis: Grounded theory after the postmodern turn*. 2005: Sage.
[41] Zeiss, R. and P. Groenewegen, *Engaging boundary objects in OMS and STS? Exploring the subtleties of layered engagement.* Organization, 2009. **16**(1): p. 81-100.
[42] Star, S.L., G.C. Bowker, and L.J. Neumann, *Transparency beyond the individual level of scale: Convergence between information artifacts and communities of practice.* Digital library use: Social practice in design and evaluation, 2003: p. 241-269.

[43] Carlile, P.R., *A pragmatic view of knowledge and boundaries: Boundary objects in new product development.* Organization science, 2002. **13**(4): p. 442-455.
[44] Jonsson, K., J. Holmström, and K. Lyytinen, *Turn to the material: Remote diagnostics systems and new forms of boundary-spanning.* Information and Organization, 2009. **19**(4): p. 233-252.
[45] Onwuegbuzie, A.J. and N.L. Leech, *Validity and qualitative research: An oxymoron?* Quality & Quantity, 2007. **41**(2): p. 233-249.
[46] Maxwell, J.A., *Qualitative research design (Vol. 41)*. 1996, Thousand Oaks, CA: Sage Publications.
[47] Becker, H.S., *Sociological work*. 1971: Transaction publishers.
[48] Kazi, S.A. *A conceptual framework for web-based intelligent learning environments using SCORM-2004*. in *Advanced Learning Technologies, 2004. Proceedings. IEEE International Conference on*. 2004. IEEE.

VIGGO HOLMSTEDT
University College of Southeast Norway
Department of Economics and Managemen
NORWAY
E-mail: vh@usn.no

SHEGAW A. MENGISTE
University College of Southeast Norway
Department of Economics and Management
NORWAY
E-mail: sme@usn.no

Fifth International Conference
Modelling and Development of Intelligent Systems
June 23 - 25, 2017
"Lucian Blaga" University
Sibiu - Romania

Using Top Down Clustering model for the assessment of anthropogenic pressures and threats on the fish populations

Daniel Hunyadi, Angela Bănăduc, Doru Bănăduc

Abstract

The study is based on the necessity of classification and hierarchization of pressures and threats on fish species, in relation with their intensity and spatial extension, as an essential step for ecological impact assessment and management. The input data of our model are represented by human pressures and threats categories on fish species present in the Lower Danube - Danube Delta - North-western Black Sea convergence area, the spatial frequency and the intensity levels of each categories varying from one (minimum) to five (maximum).

1 Introduction

The identified pressures and threats categories are: anthropogenic geo-morphological changes, anthropogenic hydrological changes, habitat fragmentation and/or loss, aquatic and semi-aquatic vegetation diminishing/disparition, pollution and/or eutrophication, fish pouching and/or overexploitation, alien species introduction, trophic resources diminishing/destruction. The obtained model is a combination of decision tree learning and clustering. We chose as being appropriate the RapidMiner (RM) for implementation of this model.

Assessing the capacity of ecosystems to respond to a huge diversity of natural and/or anthropogenous perturbations or disturbances by resisting structural and functional damage and recovering, the so called resilience [1], it is one of the most attractive intellectual problem for any student in the field of ecology and remain a challenge for all his life after, along the ongoing debates related to how to protect the aquatic ecosystems in terms of their resilience assessment and monitoring. This issue is obviously one which cannot be approached by a single point of view and/or taxa, and/or metric. The time scale is very important in this respect in the context in which the rate at which a natural system returns to a single steady or cyclic state [2] it is time dependent. The working hypothesis in this specific research that the fish fauna structure significant variability and return/recovery can be used as a comparative tool to assess diverse types of aquatic ecosystems resilience in the condition of the human impact presence.

Why Lower Danube River-Danube Delta-North West Black Sea area? This convergence area is a rich, complex and dynamic fish fauna formed of three interdependent subsystems which exhibits a significant level of flexibility and adaptation over geological time [3].

The rest of article is organized as follows. In section 2 we present some literature review regarding to clustering and decision trees. Input data, model definition, model implementation and practical results are presented in section 3. Conclusions and further directions of study can be found in section 4.

2 Literature review

The identified pressures and threats categories are: anthropogenic geo-morphological changes, anthropogenic hydrological changes, habitat fragmentation and/or loss, aquatic and semi-aquatic vegetation diminishing/disparition, pollution and/or eutrophication, fish pouching and/or overexploitation, alien species introduction, trophic resources diminishing/destruction. The obtained model is a combination of decision tree learning and clustering. We chose as being appropriate the RapidMiner (RM) for implementation of this model.

2.1 Clustering

In this section, K-means clustering algorithm and some of the widely known weighted clustering algorithms have been described in brief with their limitations.

K-means [9] is one of the most popular clustering algorithms. K-means is a partitioning method, which creates initial partitioning and then uses iterative relocation technique that attempts to improve the partitioning by moving objects from one group to another. The algorithm is used to classify a given data set into fixed number of clusters (K). K-means uses the concept of centroid, where a centroid represents the center of a cluster. In the process, K centroids, one for each cluster is defined apriori. Then each object of the data set is assigned a group with the closest centroid. The positions of k centroids are recomputed when all objects have been assigned to any of the clusters. The process is repeated until the centroids are no longer move. One of the limitations of K-means is not differentiating among attributes i.e. each attribute is given same importance in the clustering process. To overcome the limitation, weighted K-means is used with different weights for different attributes.

Al-Harbi et.al. [2] applied simulated annealing to generate weights for weighted K-means. Local optimization problem is one of the problems of simulated annealing techniques and that forms a major limitation in this approach too. Ayan et. al. [3] used information gain as attribute weights but the approach has the inherent drawbacks of the concept of information gain. A genetic cluster algorithm is also proposed by Demiriz et. al. [7] that has difficulty in defining fittest function as required in genetic process. In many real world problems, clustering in stand-alone mode does not provide the desired results.

Semi-supervised clustering [5, 6, 10, 15] is becoming popular with the presence of both labeled and unlabeled data in many practical problems. Semi-supervised clustering uses a small amount of labeled objects (where information about the groups is available) to improve unsupervised clustering algorithms. Existing algorithms for semi-supervised clustering can be broadly categorized into constraint-based and distance-based semi-supervised clustering methods. Constraint-based methods [5, 6, 10] are generally based on pair-wise constraints i.e. pairs of objects labeled as belonging to same or different clusters to facilitate the algorithm towards a more appropriate partitioning of data. In this category, the objective function for evaluating clustering is modified such that the method satisfies constraints during the clustering process. In distance-based approaches [5, 11], an existing clustering algorithm uses a particular distance measure.

Wagstaff et. al. [10] has developed another variant of the k-means algorithm i.e. COP-Kmeans by incorporating background knowledge in the form of instance-level constraints. These instance-level constraints help in identifying which objects should be grouped together. An if-statement is introduced to assign cluster and ensures that none of the constraints are violated when the k-means algorithm groups each object to its closest cluster. However, the major limitation of this algorithm is that it does not allow violation of constraints even if it leads to a more cohesive clustering and leaving them vulnerable to noisy supervision. In order to overcome this limitation, Basu et. al. [6] has proposed pair-wise constraint k-means (PCKmeans) algorithm.

PC-Kmeans algorithm is similar to COP-Kmeans, but the main difference is that this algorithm can violate the constraints with some trade off as penalty for doing so. It tries to come up with a good cluster formation while minimizing the penalty that it incurs. A major limitation of this approach is that it assumes a single metric for all clusters, preventing them from having different

shapes. Bilenko et. al. [5] has proposed metric pair-wise constraint kmeans (MPCK-Means) algorithm to get rid of this limitation. MPCK-Means is considered as one of the most popular semi-supervised clustering algorithms in the recent past. Therefore, the proposed approach has been compared with MPCK-Means in the paper. The proposed approach based on Hyperlink-Induced Topic Search (HITS) algorithm is introduced to overcome the limitations of earlier work.

2.2 Decision trees

Decision tree is a popular classification method that results in a flow - chart like tree structure where each node denotes a test on an attribute value [3] and each branch represents an outcome of the test. Decision tree is a supervised data mining technique. It can be used to partition a large collection of data in to smaller sets by recursively applying two-way and /or multi way splits.

Using the data, the decision tree method generates a tree that consists of nodes that are rules. Each [8] leaf node represents a classification or a decision. The training process that generates the tree is called induction. It has been shown in various studies that employing pruning methods can improve the generalization performance of a decision tree; a loosely stopping criterion is used, letting the decision tree to over fit the training set. Then the over-fitted tree is cut back into a smaller tree by removing sub–branches that are not contributing to the generalization accuracy.

Step 1: Create a node N
Step 2: If samples are all of the same class, C then
Step 3: Return N as a leaf node classify with the class C;
Step 4: If attribute-list is empty then
Step 5: Return N as a leaf node classify with the most common class in samples.
Step 6: Select test-attributes, the attribute among attribute-list with the highest information gain;
Step 7: Label node N with test-attribute;
Step 8: For each known value ai of test-attribute.
Step 9: Grow a branch from node N for the position test attribute = ai;
Step 10: Let Si be the set of samples for which test-attribute = ai;
Step 11: If Si is empty then
Step 12: Attach a leaf classify with the most common class in samples;
Step13: Else attach the node returned by generate-decision tree (Si, attribute-list-attribute);

Each internal node tests an attribute, each branch corresponds to characteristic value, and each leaf node assigns a classification.

3 Model specification

3.1 Input data

The input data of our model are represented by human impact pressures and threats categories presence in Lower Danube (1), Danube Delta (2) and Northwestern Black Sea (3) convergence area, and of the human impact pressures and threats on the fish with quantitative effects, with their intensity level varying from 1 (minimum) to 5 (maximum). The identified pressures and threats categories are: anthropogenic geomorphological changes (A), anthropogenic hydrological changes (B), habitat fragmentation and/or loss (C), aquatic and semiaquatic vegetation diminishing/disparition (D), pollution and/or eutrophication (E), fish pouching and/or overexploitation (F), alien species introduction (G), trophic resources diminishing/disparition (H).

These variables were measured for 115 species. Data were centralized in an EXCEL document.

3.1 Model definition

Our model is built in several steps. In the first step, pre-processing data step, input data are normalized in order to be ready for processing. Then, we use divisive hierarchical clustering in order to obtain the optimum number of clusters.

Normalized input data and the optimum number of clusters are used by k-means algorithm. This algorithm is one of the simplest unsupervised learning algorithms that solve the clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed apriori. The main idea is to define k centers, one for each cluster. These centers should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest center. In the last step, we obtained our model which is a combination of decision tree learning and clustering).

3.1 Model implementation and practical results

We chose RapidMiner (RM) for implementation of our model. For the hierarchical clustering we use *TopDownClustering* operator. *KMeans* operator solve the non- hierarchical clustering task offering as output data the clusters. The representation of the clustering solution as a decision tree is realized using *DecisionTree* operator. The implemented processes and the practical results are presented in the next section.

First process is used in order to obtain the optimum number of clusters. It use an import operator named *ReadExcel* which reads an *ExampleSet* from the specified Excel file. The *NominalToNumerical* operator is used for pre-processing data named which changes the type of selected non-numeric attributes to a numeric type.

The *TopDownClustering* operator is used for hierarchical clustering and performs top down clustering by applying the inner flat clustering scheme recursively. Top down clustering is a strategy of hierarchical clustering. The result of this operator is a hierarchical cluster model. The output data for this process is the optimal number of clusters. As we can see in figure 1, for our input data we obtained 19 clusters.

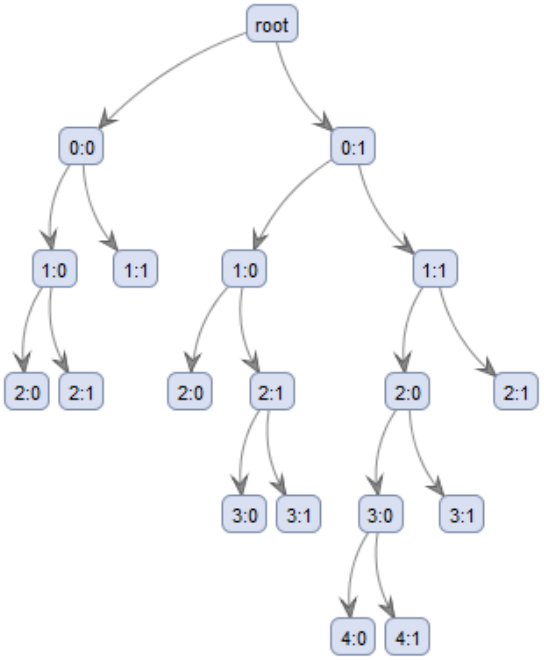


Figure 1. Output data of hierarchical clustering

Second process is a combination of clustering and decision tree and is used to obtain the representation of the model. The chain of the process use *KMeans* operator which performs clustering using the *k-means* algorithm. This operator contains a parameter which specifies the number of clusters to form. The input value for this parameter is the value obtained in the first process.

The *ChangeAttributeRole* operator is used to change the role of one or more attributes. The Role of an attribute reflects the part played by that attribute in an *ExampleSet*. Changing the role of an attribute may change the part played by that attribute in a process. One attribute can have exactly one role. The target role for out attribute is label.

The final operator in our chain is *DecisionTree*. This operator generates a decision tree for classification of both nominal and numerical data. A decision tree is a tree-like graph or model. It is more like an inverted tree because it has its root at the top and it grows downwards. This representation of the data has the advantage compared with other approaches of being meaningful and easy to interpret.

The chains of the process is presented in figure 2 and the clusters obtained are presented in figure 3.

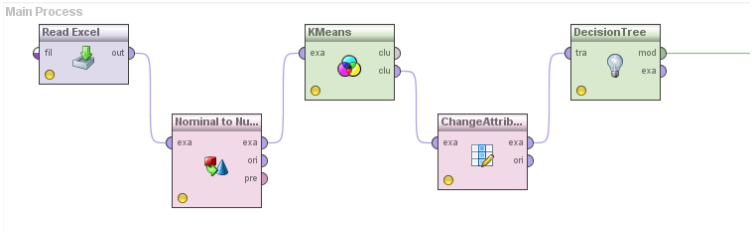


Figure 2. Non-hierarchical process

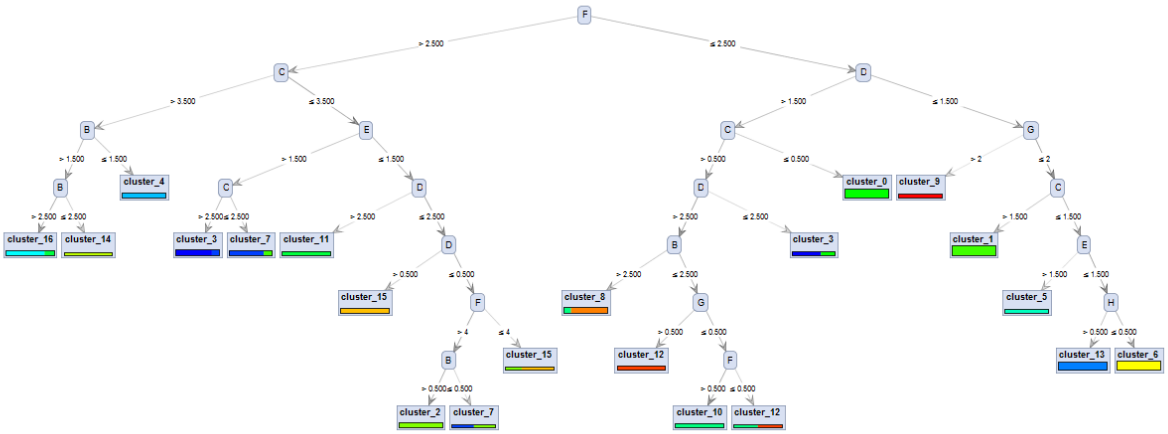


Figure 3. Decision Tree

As we can see in figure 3, the obtained model highlight the fact that the F (fish pouching and/or overexploitation) human impact category has an overall influence on 100% of the studied fish species, in river, delta and marine environments. The studied area was influenced by the human interventions inducing powerful environment alterations, these interventions consisted in large areas embankments for extensive agriculture, intensive fish culture and forestry, with effects on the natural processes, on ecological equilibrium and on the wetlands functions.

Decision trees are powerful and popular for both classification and prediction. The attractiveness of tree-based methods is due largely to the fact that decision trees represent rules. A decision tree model consists of a set of rules for dividing a large heterogeneous population into

smaller, more homogeneous groups with respect to a particular target variable. The intermediate nodes represents the categories for impact pressures. The leaf are the clusters where ale species are segmented. The values from the edges represents the rules for dividing.

4 Conclusions

Based on the 115 fish species and six human impact categories (A - anthropogenic geo-morpholgical changes, B - anthropogenic hydrological changes, C - habitat fragmentation and/or loss, D - aquatic and semi-aquatic vegetation diminishing/disparition, E - pollution and/or eutrophication, F - fish pouching and/or overexploitation, G - alien species introduction, H - trophic resources diminishing/destruction) analysis, the obtained model reveal the connections among some of the human impact categories and variable number of fish.

It is hard to define a mathematical approach to attribute space analysis and description for classification model building because the efficiency of methods and their parameters depend on data structure. Clustering-based decision tree classifier construction ent results and fit certain data structures. Hierarchical clustering is a suitable approach to high density area discovery within classes because of its elasticity choosing cluster defining parameters (similarity measures and linkage options) and it also does not need any prior information about the number of clusters and their positions (centroids). This allows a more objective class structure analysis. The choice of classification model is also relevant because the best model for initial data is not always the best model for data with decomposed classes. This also depends on class structure and the character of overlapping areas. There cannot be a universal approach to cluster combination selection without taking into account the final classification

One possible direction of our study consists in using our proposed methods for other case studies to see their level of generality.

References

[1] Holling, C., 1973. Resilience and stability of ecological systems. Annu Rev Ecol Syst 4:1-23.
[2] Holling, C., 1996. Engineering resilience versus ecological resilience. In Schultze P editor. Engineering within ecological constraints. Washington DC National Academy. 31-44.
[3] Bănăduc D., Planellas S., R., Trichkova T., Curtean-Bănăduc A., 2106. The Lower Danube River-Danube Delta-North West Black Sea: A pivotal area of major interest for the past, present and future of its fish fauna – A short review. Science of The Total Environment, 545-546: 137-151, DOI: 10.1016/j.scitotenv.2015.12.058
[4] S. H. Al-Harbi and V. J. Rayward-Smith, “Adapting k-means for supervised clustering,” Applied Intelligence, vol. 24, pp. 219–226, 2006. [17] W. M. Rand, “Objective criteria for the evaluation of clustering methods,” Journal of the American Statistical Association, 66, 846- 850, 1971.
[5] F. Ayan, “Using information gain as feature weight,” in Proc. of 8th Turkish Symposium on Artificial Intelligence and Neural Networks (TAINN’99), Turkey 1999.
[6] S.Balaji and Dr.S.K.Srivatsa” Decision Tree induction based classification for mining Life Insurance Data bases” International Journal of Computer Science and Information Technology & Security (IJCSITS), ISSN: 2249-9555 Vol. 2, No.3, June 2012 pp-699-703.
[7] M. Bilenko, S. Basu and R. Mooney, “Integrating constraints and metric learning in semi-supervised clustering,” in Proc. ICML, 2004, pp. 81–88.
[8] S. Basu, A. Banerjee, and R. J. Mooney, “Active semi-supervision for pairwise constrained clustering,” in Proc. of the 2004 SIAM International Conference on Data Mining (SDM-04), 2004.
[9] A. Demiriz, K. P. Bennett, and M. J. Embrechts, “Semi-supervised clustering using genetic algorithms,” Artificial Neural Networks in Engineering (ANNIE), (1999) 809–814.
[10] Lior Rokach and Oded Maimon” Top-Down Induction of Decision Trees Classifiers- A Survey” IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS: PART C, VOL. 1, NO. 11, NOVEMBER 2002 pp- 1-12
[11] J. McQueen, Some methods for classification and analysis of multivariate observations, Proc. Symp. Math. Statist. And Probability, 5th, Berkeley, 1 (1967) 281–298.

[12] K. Wagstaff and S. Rogers, "Constrained k-means clustering with background knowledge," in Proc. of 18th International Conference on Machine Learning, 2001, pp. 577–584.
[13] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell, "Distance metric learning, with application to clustering with side information," Advances in Neural Information Processing Systems, vol. 15, pp. 505–512, 2003.

Daniel Hunyadi
"Lucian Blaga" University of Sibiu
Department of Mathematics and Informatics
5-7 Dr. Ratiu Street 550012
Romania
E-mail: daniel.hunyadi@ulbsibiu.ro

Angela Bănăduc
"Lucian Blaga" University of Sibiu
Department of Environmental Sciences, Physics, Physical Education and Sport
5-7 Dr. Ratiu Street 550012
Romania
E-mail: angela.banaduc@ulbsibiu.ro

Doru Bănăduc
"Lucian Blaga" University of Sibiu
Department of Environmental Sciences, Physics, Physical Education and Sport
5-7 Dr. Ratiu Street 550012
Romania
E-mail: doru.banaduc@ulbsibiu.ro

Fifth International Conference
Modelling and Development of Intelligent Systems
June 23 - 25, 2017
"Lucian Blaga" University
Sibiu - Romania

Factors Influencing Group Creativity of Computer Science Students: A Fuzzy Cognitive Map

Gabriela Moise, Monica Vladoiu, Zoran Constantinescu

Abstract

Educational paradigms transform constantly to stay tuned with the change in our society. Promoting collaboration and boosting creativity in learning are major trends today. Therefore, increasing creative and collaborative skills of both students and employees is currently of immense interest for stakeholders in education, industry, policy making etc. In this paper, we first overview the main factors shown in the literature as having an influence on group creativity. Then, we approach the construction of the most (optimally) creative groups given a cohort of students and a particular learning scenario, based on the influence of various factors on group creativity. Our method is based on using fuzzy cognitive maps to capture the influence of these factors on group creativity, which are built accordingly with the results in the literature, our experience, and empirical data obtained during the instructional activities of our Software Engineering class. However, the method is general and it can be adapted to any learning scenario in any domain. A procedure on using this method is available as well.

1 Introduction

Increasing creative and collaborative skills of both students and employees is currently of immense interest for stakeholders in education, industry, policy making etc. Despite the abundance of research on factors that influence creativity of individuals and groups or teams, only few works mention metrics that express dependency rates between these factors and creativity [1-5]. When it comes to creativity, we acknowledge the difference between *group* and *team*. While both terms represent multiple people working together towards a common objective, in the case of teams collaboration relationships between the members have been established over longer periods of time compared to groups [2]. Furthermore, as a consequence of the shared work past, the team members have developed similar interests and affinities towards certain subjects. Moreover, often they share the same values.

Learning groups are working groups that evolve during common educational scenarios that unfold over long periods of time and, generally, become teams, based on the evolution of the relationships inside the group. Creativity of learning groups can be approached within augmented collaborative learning environments in which student learning groups work creatively, both at individual and social level, to fulfill particular tasks, to complete specific projects, or to achieve some particular goals. The results of their work can be problem solutions, papers, overviews, (pieces of) software or hardware, documents, essays, etc. The degree of creativity of these results is evaluated by instructors and, this way, a measurement of group creativity can be obtained. An

example of an augmented collaborative learning environment can be a classroom with instructional materials and/or equipments (e.g. drawings, robots, drones, maps etc.), along with a set of teaching and learning methods (problem-based learning, brainstorming, project-based learning, game-based learning, etc.) that stimulate imagination, creativity, and innovation.

The focus of this paper is dual, first to overview the main factors shown in the literature to have an influence on group creativity and, second, to present our work on using fuzzy cognitive maps to capture the influence of these factors on group creativity. The fuzzy cognitive map that we have constructed is based on the results in the literature, our experience, and empirical data obtained while working with Computer Science students enrolled in our Software Engineering course. However, the method is general and it can be adapted to any learning scenario in any domain. A procedure on using this method for a given cohort of students and a particular learning scenario is available as well.

The structure of this paper is as follows: the next section includes the related work, the third one presents fuzzy cognitive maps, the fourth one introduces our work on using fuzzy cognitive maps to capture the influence of various factors on group creativity within learning scenarios, while the last section include the conclusions and some future work ideas.

2 Factors that Influence Creativity of Groups

In this section, we overview the related work on group/team creativity and on creativity during working and learning situations. In [6], the authors have analyzed the cause-effect relationships between 6 factors: team creativity, exploitation, exploration, organizational learning culture, knowledge sharing, and expertise heterogeneity. The main research issue addressed in this work was *how do the processes of creative revelation—exploitation and exploration—engaged in by team members contribute to building team creativity, and how do environmental factors—organizational learning culture, knowledge sharing, and expertise heterogeneity—affect team creativity*. A general Bayesian Network of the dependencies between these factors and team creativity have been used within scenario-based simulations to show that a direct relationship exists between team creativity and exploitation, exploration, organizational learning culture, knowledge sharing, and expertise heterogeneity. Also, exploration is correlated to organizational learning culture and exploitation is associated with expertise heterogeneity. Moreover, to sustain high levels of team creativity both organizational learning culture and knowledge sharing are ought to remain high [6]. Team creativity is influenced by a variety of team characteristics such as size, Skills, Knowledge and Abilities (SKAs), diversity (age, gender, ethnicity), psychological and participative safety, leadership, conflict or cohesion groups, and group confidence [2]. In [3], the authors raise an interesting issue related to group creativity, namely the tendency toward conformity, and propose inclusion of new members as a coping mechanism that further stimulates innovation.

Cultivation and promotion of creativity are highly sought after in Higher Education and personalized learning and game-based learning are seen as important ways of acquiring these goals in [7]. A model of collaborative creativity that takes into account four categories of variables and three categories of processes which influence creativity and innovation is provided in [8]. The four categories of variables are group member variables, group structure, group climate, and external demands, while the three categories of processes are cognitive, motivational, and social. Learners' creativity can be triggered by several factors such as awareness of creativity's role within our society and in everyday life, development of social skills, using critical thinking models, encouraging brainstorming sessions followed by questions' and answers' sessions, involvement in multicultural or multidisciplinary tasks, etc. [9, 10].
Strong dependencies between learning styles and creativity results from a study presented in [11]. Identification of the relationship between learning styles and learners' creativity is researched

actively in educational psychology because it could help with identification of correct guidance and careful planning for motivating learners to develop and adopt appropriate pedagogical models [11]. A model for evaluation of activities that cultivate creative skills and attitudes, which can be used during planning of educational processes, is provided in [12]. Three main categories of indicators are taken into account, i.e. *cognitive category* (the student's abilities to reason on the content at hand, to make connections between existing elements, to create hypotheses and to construct new meanings while accomplishing the proposed task); *affective category* (that shows how much students like and value what they learn and how much they engage in the proposed activity and that also reflects their emotional status, behaviors, and attitudes they show while working on their task), and *meta-cognitive category*, which illustrates students' ability to take the overall process under control either during or at the end of the learning activity [12].

During the eighties, Amabile has developed *The Componential Model of Creativity* for individual creativity, which she has further extended to team creativity and innovation in organizations [13, 14]. Building on her previous work, she also proposed a componential theory of creativity which includes *three within-individual components* (*domain-relevant skills*, *creativity-relevant processes*, *task motivation*) and a component outside the individual, i.e. the *social environment* [15]. This theory emphasizes that creativity calls for a convergence of all these and that *creativity should be highest when an intrinsically motivated person with high domain expertise and high skill in creative thinking works in an environment highly supporting creativity*.

There is still a lot of controversy in the literature about the way in which some factors influence creativity, i.e. positively or negatively. For example, in [5], group cohesion is generally seen as positive, but it can also lead to rejection of criticism and less critical thinking, resulting in lower creativity and innovation. Moreover, for service oriented teams, task conflict has shown no effect on team creativity, while relationship conflict was *significantly and negatively related to team creativity* [5]. For other type of teams that focus on technology projects, no effects were determined for relationship conflict, while *task conflict was strongly associated with increased creativity* [5]. Other factors are considered as well in the literature, but so far the results are non conclusive – for example, with regard to group diversity, some studies show its positive effects, other show the opposite, while some find no effect whatsoever [2].

3 About Fuzzy Cognitive Maps

Our approach consists in using Fuzzy Cognitive Maps (FCMs) to analyze the influence of various factors on group creativity. *FCMs are fuzzy-graph structures for representing causal reasoning* [16]. FCMs derive from both cognitive maps and fuzzy logic and capture the dynamic of modifications within systems. Axelrod has introduced Cognitive Maps (CMs), as digraphs with the vertices representing *concept variables* and the arcs showing *the causal relations* between the concepts (with two possible values -1 or 1) [17]. The value +1 associated with an arc from the vertex A to the vertex B shows that A causally increases B, while -1 shows that A causally decreases B. CMs are represented with adjacency matrices having elements -1, 1 or 0. The value 0 signifies that there is no arc between the respective vertices, i.e. no causality between the respective concepts exist. Kosko has extended the cognitive maps allowing that values on the arcs belong to the interval [-1, 1] and iteratively computed the influence of a factor (vertex) on other factor (vertex) using neural networks-based methods [16]. The values associated with the arcs are causal values and can be defined by fuzzy values. Considering the following linguistic terms for causal values *{very low<=low<=none<=some<= high<=very high}* associated to links between nodes, a FCM with 4 concepts is represented in Fig. 1.

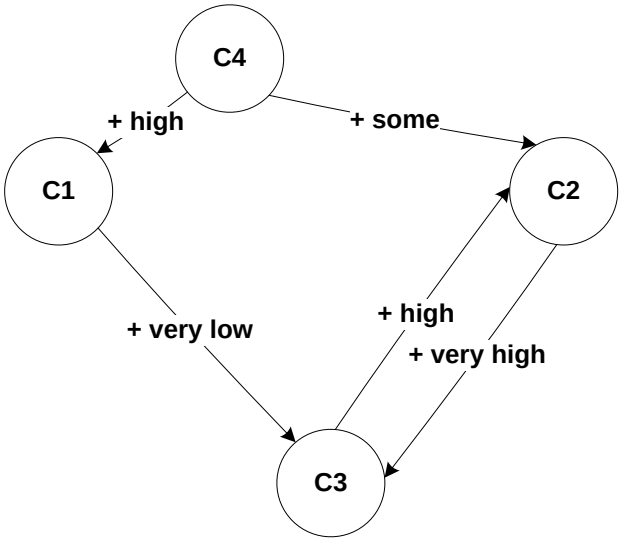


Figure 1: A Fuzzy Cognitive Map with 4 Concepts

To determine how the modification value of the concept C4 will affect the value associated to the concept C3, the technique presented in [16] can be used. Thus, there are two causal paths between concepts C4 and C3: (C4, C2, C3) and (C4, C1, C3).

The indirect effect through (C4, C2, C3) is:

$$min\{some, very\ high\} = some. \quad (1)$$

The indirect effect through (C4, C1, C3) is:

$$min\{high, very\ low\} = very\ low. \quad (2)$$

The total effect of C4 on C3 is:

$$max\{some, very\ low\} = some. \quad (3)$$

FCM can be also seen as a type of *Recurrent Artificial Neural Network (RANN)* with learning capacity [20, 21]. In this case, the values associated to arcs are called weights and take values in the interval [-1,1]. A FCM containing 4 concepts is presented in Fig. 2.

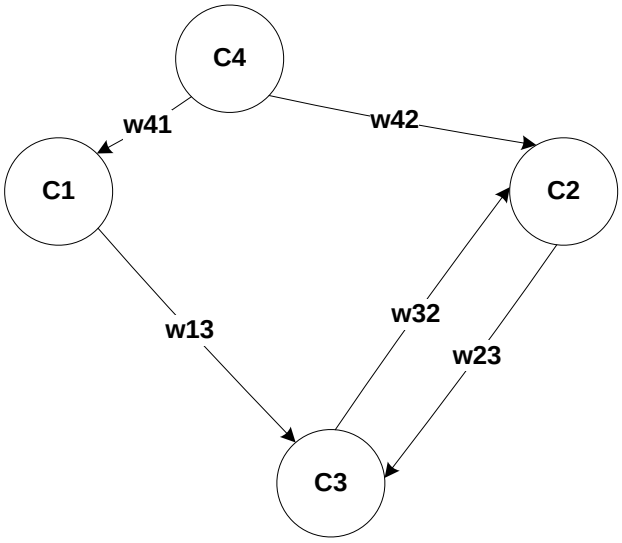


Figure 2: A Fuzzy Cognitive Map (as RANN) with 4 Concepts

The concepts are represented using vertices and the arcs between vertices show the dependencies between the respective concepts.

- A positive weight ($w_{ij}>0$) means that an increasing of the value of a concept C_i will determine an increasing of the value of the concept C_j;
- A negative weight ($w_{ij}<0$) means that an increasing of the value of a concept C_i will cause a decreasing of the value of the concept C_j;
- A null weight means that there are no dependencies between the concepts C_i and C_j.

The concepts' values are calculated using formula 4.

$$c_i(t) = f\left(\sum_{\substack{j=1 \\ j\neq i}}^{n} c_j(t-1)w_{ji}\right) \quad (4)$$

where n represent the number of concepts (in our case n=4), $c_j(t)$ is the value associated to the concept C_j, f is a transfer function, and w_{ji} is the weight of the link between C_j and C_i.

The most used transfer functions in FCMs are the sign function, the trivalent function, or the sigmoid function [20, 21]. More information on FCMs can be found in [16-21].

FCMs can be built using human expertise or using training data sets and a learning algorithm. The FCM that captures the dependencies between various factors and group creativity in this work has been built based on the results in the literature, our experience, and empirical data obtained while working with Computer Science students enrolled in particular course.

4 A Fuzzy Cognitive Map on Group Creativity in Computer Science Higher Education

A general FCM that represents the dependencies between group creativity and some factors identified in the related work is shown in Fig. 3. The arc between group creativity and learning style is purposely left unlabelled because no established correlation is available yet and, moreover, tackling this issue is very difficult given the variety of learning styles (visual, auditory, kinesthetic, etc.). The group dimension has to be relatively small to have a positive influence on creativity, but not to small – for example, a group of five will generally be more creative than a group of two. Some factors may have a negative influence on group creativity, for example, too much or too little controversial communication or task conflict [5]. The biggest challenge of building a FCM for group creativity consists in determining each value associated to each arc between a specific factor and group creativity.

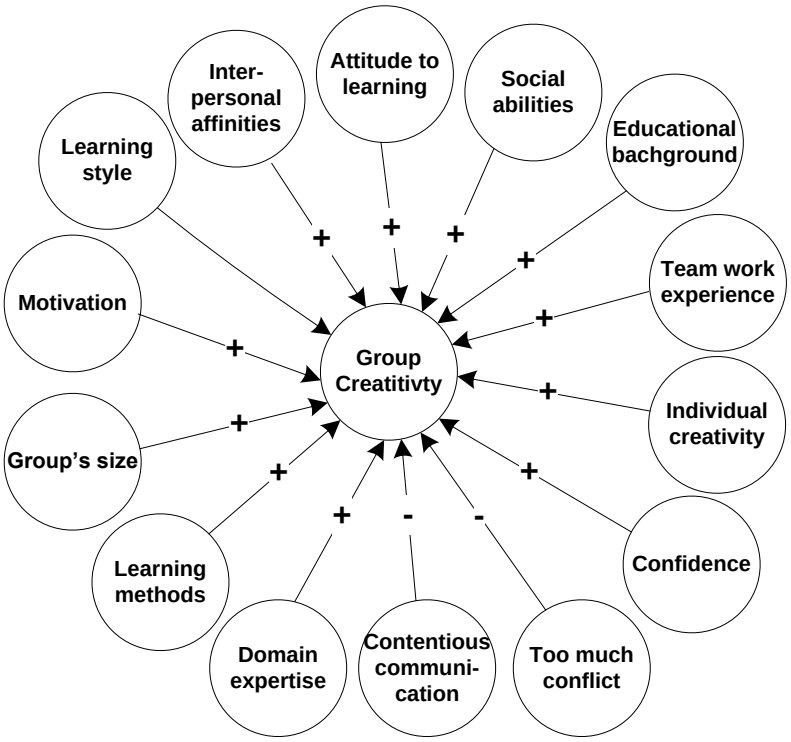


Figure 3: A FCM for Group Creativity

One of our real-world experiments aiming at building such a FCM was performed two years ago on 20 of our final-year Computer Science students enrolled in the Software Engineering course. The final grade measures both how well they have achieved the course requirements with respect to the domain knowledge and how well they work together in small developers' teams that need to complete a common software development project and to present properly their work. For our first Group Creativity – Fuzzy Cognitive Map (GC-FCM) we have considered only four factors: *individual creativity, motivation, domain expertise, and inter-personal affinities*. We aim to continue this work and to consider more and more factors, and to evaluate the degree in which each such factor influences group creativity.

First, we performed a Gough-based evaluation of creativity of our students [25]. In general, the range of individual creativity on the Gough scale is [-12, +18]. The creativity score mean in our case is 2.55 (Table 1). The values obtained for the two other attributes considered in the classification are shown in Table 1 (the students are distinguished by their unique identifier). The domain expertise is the grade obtained at the Data Structures and Algorithms class, while they were sophomores. We have chosen this grade because the programming part of the Software Engineering project consists of developing Java computer applications with fundamental data structures and algorithms. The motivation attribute has been determined using a questionnaire based on MSLQ that we have adapted for Computer Science students. MSLQ is a multi-item, self-report Likert-scaled instrument designed to assess motivation and use of learning strategies by college students [22]. A value of 2 for motivation means a highly motivated student, a value of 1 means a motivated student, while a value of 0 means a less motivated student.

Table 1: Individual creativity factors of final-year students

	Gough score	Domain Expertise	Motivation
Learner 1	5	8	2
Learner 2	4	8	1
Learner 3	7	8	2
Learner 4	7	10	2
Learner 5	8	8	1
Learner 6	3	8	2
Learner 7	2	7	0
Learner 8	2	6	0
Learner 9	2	6	1
Learner 10	-2	5	0
Learner 11	8	10	1
Learner 12	-2	7	1
Learner 13	-1	6	2
Learner 14	7	7	1
Learner 15	4	8	1
Learner 16	0	5	2
Learner 17	5	5	2
Learner 18	3	5	0
Learner 19	-5	6	0
Learner 20	-6	6	0

During this real-world scenario, the students have grouped themselves in small teams based on their inter-personal affinities (the members of each teams were buddies). Four cliques resulted this way (the numbers between parentheses are student identifiers), namely Group 1 (1, 2, 3, 4, 5, 6), Group 2 (7, 8, 9, 10), Group 3 (11, 12, 13, 14, 15, 16, 17), and Group 4 (18, 19, 20).

When the group creativity was measured (by evaluating their projects with respect to meeting the requirements, including creativity), we obtained the following results:

- no group was in the high creativity class (H);
- two groups (groups 1 and 3) pertained to the medium creativity class (M);
- two groups (groups 2 and 4) belonged to the low creativity class L.

To determine the values associated to relationships between group creativity and the influencing factors two methods are available, i.e. an expert-based method and a data-based method [23]. In the expert-based method, each expert determines the influence of the factors on group creativity using linguistic values (such as low, very high, strong, very strong etc.) and all these linguistic values are combined using an aggregation function. The data-based method is more elaborated - it uses a FCM learning algorithm and training data. The process of obtaining the training data takes time and well formulated procedures to measure the factors using numerical values are necessary. More information about learning algorithms can be found in [24].

Based on the empirical data resulted from our first experiments with Computer Science students enrolled in our Software Engineering course, our experience (using human expertise being a method to construct FCMs), and some results in the related work [15], we have got an estimation, based on mathematical mean, for the influence of domain expertise, individual creativity, motivation, and inter-personal affinities on group creativity. Thus, all these factors have a positive influence on creativity. However, the relation below may be true only for the cohorts of students

that we have experimented with (we repeated the experiment two years in a row). Nevertheless, we believe that it can be a starting point for further experiments with Computer Science students.

---

grade of influence of *domain expertise*>= grade of influence of *individual creativity*>= grade of *influence of motivation*>= grade of influence of *inter-personal affinities*

Given the large number of factors affecting group creativity, we propose using the linguistic values in [23] to assess the influence degree, namely *negatively very strong*, *negatively strong*, *negatively medium*, *negatively weak*, *negatively very weak*, *zero*, *positively very weak*, *positively weak*, *positively medium*, *positively strong*, *positively very strong*, and *positively very very strong*. Based on the relation above and using these possible values for the influence degree, we have constructed a particular group creativity FCM for Computer Science students (Fig. 4).

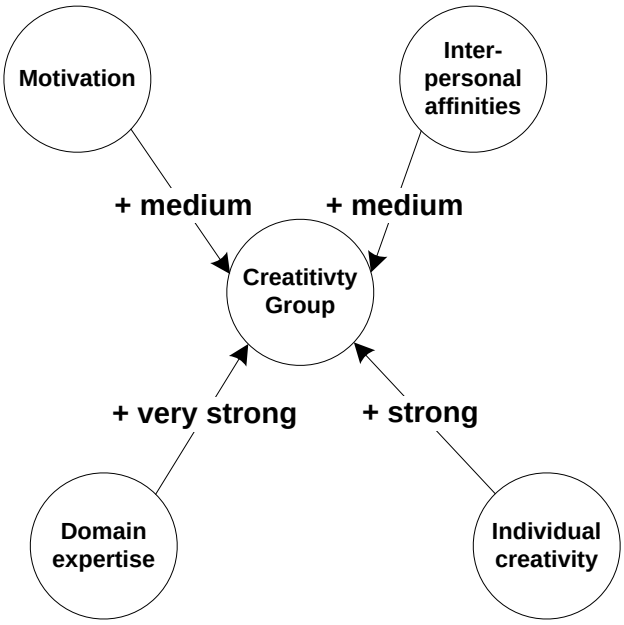


Figure 4: The resulted FCM of Group Creativity

This FCM can be further used to experiment with building groups of students and making predictions about their creativity. In addition, comparative estimations of the influence that various factors have on creativity can be performed because group creativity depends on both individual factors (and their influences) and interactions that take place between group members.

To use this method, one needs follow the procedure beneath that starts by establishing the factors that influence group creativity to be taken into account, followed by building a FCM with help from experts in the educational domain of interest (in our case, Computer Science education). The obtained FCM can be further improved using machine learning algorithms. The refined FCM can be used for making predictions about group creativity given a cohort of students and a particular learning scenario. A software tool that implements the machine learning part can be integrated as well. For example, a FCM tool for Matlab already exists and can be used to develop models and to obtain predictions on group creativity [26].

Procedure for construction of a particular FCM for a cohort of students and a learning scenario

1. Establish the factors that influence group creativity to be considered
2. Build a FCM based on human expertise in the field of interest
3. Refine the FCM obtained in step 2 using machine learning algorithms and specific tools
4. Use the refined FCM to make predictions on group creativity given a cohort of students and a particular learning scenario

5 Conclusions

Educational paradigms change constantly to stay tuned with evolution of our society and, consequently, promoting collaboration and boosting creativity in learning are major trends today. This paper approached the construction of "the most" (optimally) creative groups, given a cohort of students and a particular learning scenario, and taking into account various factors that influence creativity, both at individual and group level. This is not an easy task, as the related work shows. Some factors positively influence group creativity (such as domain expertise), some others may have a positive influence given that they are in an appropriate amount (such as controversial communication or task conflict) and are correlated with the type of activities that groups perform (for example, service-oriented or technology development), while the influence or others is non-conclusive (e.g. group diversity). The approach taken here consists in using fuzzy cognitive maps to illustrate the influence that various factors have on creativity within various learning scenarios. We have built such a map based on the empirical data resulted from our first experiments with Computer Science students enrolled in our Software Engineering course, our experience (using human expertise being a method to construct FCMs), and some results in the related work. Nevertheless, the method is general and it can be adapted to any learning scenario in any domain. A procedure on using this method for a certain cohort of students and a particular learning scenario has been included in this paper as well.

This is work in progress and many future work directions unfold. One would be to determinate the particular values for the arc weights in our FCM that correspond to particular learning scenarios. To accomplish that, more learning scenarios need to be considered in Computer Science education, as well as in other domains. A software tool that provide for construction of FCMs given a set of influence factors would be useful to facilitate the use of this method. Despite the promising results so far, our approach here is not to be used exclusively, but in combination with others that allow using numeric values for some factors that influence creativity in order to obtain the most appropriate organization of students in creative groups, in any given learning scenario.

References

[1] B. A. Hennessey and T. M. Amabile. Creativity. *Annual Review of Psychology*, 61: 569–98, 2010.
[2] P. B. Paulus, M. Dzindolet, and N. W. Kohn. Collaborative Creativity—Group Creativity and Team Innovation. In *Handbook of Organizational Creativity*, Michael D. Mumford (Eds.). Elsevier, 327-357, 2011.
[3] P. B. Paulus, H Coskun. Group Creativity. In *Encyclopedia of Creativity*, Mark A. Runco and Steven R. Pritzker (Eds.). Elsevier, 575-580, 2011.
[4] A. Kenny. 'Collaborative creativity' within a jazz ensemble as a musical and social practice. *Thinking Skills and Creativity* 13: 1–8, 2014.
[5] R. Reiter-Palmon, B. Wigert, and T. de Vreede. Team Creativity and Innovation: The Effect of Group Composition, Social Processes, and Cognition. In *Handbook of Organizational Creativity*, Michael D. Mumford (Eds.). Elsevier, 295-326, 2012.

[6] D. Y. Choi, K. C. Lee , Y. W. Seo. Scenario-Based Management of Team Creativity in Sensitivity Contexts: An Approach with a General Bayesian Network. In *Digital Creativity, Individuals, Groups, and Organizations*, Kun Chang Lee (Eds.), Integrated Series in Information Systems, Vol. 32. Springer, New York, 9-113, 2013.
[7] C. F. Lin, Yu-Chu Yeh,Y. H. Hung, R. I Chang. Data mining for providing a personalized learning path in creativity: An application of decision trees. *Computers & Education*, 68: 199-210, 2013.
[8] P. B. Paulus, & M.Dzindolet. Social influence, creativity and innovation. *Social Influence*, 3: 228–247, 2008.
[9] G. Moise. Triggers for Creativity in CSCL. In *Proceeding of the 9th International Scientific Conference eLearning and Software for Education*, Bucuresti, 326-331, 2013.
[10] G. Moise. Fuzzy Enhancement of Creativity in Collaborative Online Learning. In *Proceeding Revised Selected Papers on New Horizons in Web Based Learning*, Dickson K. W. Chiu, Minhong Wang, Elvira Popescu, Qing Li, Rynson Lau (Eds.). Lecture Notes in Computer Science, Vol. 7697. Springer-Verlag, Berlin, 290-299, 2012.
[11] K. A. Eishani, E. A. Saa'd , Y. Nami. The Relationship Between Learning Styles And Creativity. *Procedia - Social and Behavioral Sciences*, 114: 52-55, 2014.
[12] M. Ott, F. Pozzi. Towards a Model to Evaluate Creativity-oriented Learning Activities. *Procedia Social and Behavioral Sciences*, 2: 3532–3536, 2010.
[13] T. M. Amabile. Social Psychology of Creativity: A Componential Conceptualization. *Journal of Personality and Social Psychology*, 45, 2: 357-377, 1983.
[14] T. M. Amabile. A Model of Creativity and Innovation in Organizations. *Research in Organizational Behavior*, 10: 123-167, 1988.
[15] T. M. Amabile. Componential Theory of Creativity. Working paper, http://www.hbs.edu/faculty/Publication%20Files/12-096.pdf, 2012.
[16] B. Kosko. Fuzzy Cognitive Maps. *International Journal Man-Machine Studies*, 24: 65-75, 1986.
[17] R. Axelrod. *Structure of Decision: The Cognitive Map of Political Elites*, Princeton University Press, 1976.
[18] U. Özesmi, S. L. Özesmi. Ecological models based on people's knowledge: a multi-step fuzzy cognitive mapping approach. *Ecological Modeling*, 176, 1-2: 43–64, 2004..
[19] J. A. Dickerson and B. Kosko. Virtual Worlds as Fuzzy Cognitive Maps, *Presence: Teleoperators and Virtual Environments*, 3, 2: 173-189, 1994.
[20] C. D. Stylios and P. P. Groumpos. Mathematical Formulation of Fuzzy Cognitive *Maps, In Proceedings of the 7th Mediterranean Conference on Control and Automation (MED99)*. Haifa, Israel, 2251-2261, 1999.
[21] A. K. Tsadiras. Inference using Binary, Trivalent and Sigmoid Fuzzy Cognitive Maps, *Information Sciences*, 178, 20: 3880-3894, 2008.
[22] P. R. Pintrich, D. A. Smith, T. Garcia, W. J. Mckeachie. Reliability and Predictive Validity of the Motivated Strategies for Learning Questionnaire (Mslq). *Educational and Psychological Measurement*, 53, 3: 801-813, 1993.
[23] E. I. Papageorgiou. Methods and Algorithms for Fuzzy Cognitive Map-based Modelling. In *Fuzzy Cognitive Maps for Applied Sciences and Engineering*, Elpiniki I. Papageorgiou (Ed.). Springer, 1-28, 2014.
[24] E. Papageorgiou. Learning algorithms for fuzzy cognitive maps: A review study, *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, 2, 150-163, 2011.
[25] H. G. Gough. A Creative Personality Scale for the Adjective Check List, Journal of Personality and Social Psychology, 37:, 1398-1405, 1979.
[26] ***. 2017, FCM Research and Development Group, FCM tool, http://epapageorgiou.com/index.php/software-tools/fcm-tool

GABRIELA MOISE
UPG University of Ploiesti
CerTIMF Research Center, Department of Computer Science, Information Technology, Mathematics and Physics (iTIMF),
Bd. Bucuresti 39, Ploiesti
Romania
E-mail: gmoise@upg-ploiesti.ro

MONICA VLADOIU
UPG University of Ploiesti
CerTIMF Research Center, Department of Computer Science, Information Technology, Mathematics and Physics (iTIMF),
Bd. Bucuresti 39, Ploiesti
Romania
E-mail: mvladoiu@upg-ploiesti.ro

ZORAN CONSTANTINESCU
UPG University of Ploiesti
CerTIMF Research Center, Department of Computer Science, Information Technology, Mathematics and Physics (iTIMF),
Bd. Bucuresti 39, Ploiesti
Romania
E-mail: zoran@upg-ploiesti.ro

Fifth International Conference
Modelling and Development of Intelligent Systems
June 23 - 25, 2017
"Lucian Blaga" University
Sibiu - Romania

Improving of e-business activities by building web applications with integrated data mining services

Mircea-Adrian Mușan, Iuliana-Maria Cândea

Abstract

Our work presents a way of integrating data mining techniques, written through Rapid Miner processes, into Web applications, from a marketing perspective for business efficiency. Through the application presented in this paper we outline the advantages of integration of data mining techniques into e-commerce systems, as a new element in the progress of using informatics into e-business activities.

1 Introduction

Data mining techniques used in computer applications for e-business activities opened new possibilities for handling information in real time. Informatics systems based on these techniques assist successfully entrepreneurs in making decisions to achieve a higher degree of economic efficiency at their organisation.

Data mining techniques offers a broad and useful perspective in developing and using information systems in the field of e-business. Diverse areas and purposes of use, together with the need to remote access, are elements of departure in this work.

Data mining is the process of extracting patterns from large data sets, by combining methods from statistics and artificial intelligence with database management [1][2]. With recent tremendous technical advances in processing power, storage capacity, and inter-connectivity of computer technology, data mining is seen as an increasingly important tool by modern business, to transform unprecedented quantities of digital data into business intelligence giving an informational advantage. It is currently used in a wide range of profiling practices, such as marketing, surveillance, fraud detection, and scientific discovery. The growing consensus that data mining can bring real value has led to an explosion in demand for novel data mining technologies [3].

It is difficult to formulate one single definition for data mining. In the Figure 1 we tried to extract more equivalent definitions. Most common significations for Data Mining are "knowledge-discovery in databases" (KDD) or "extraction of interesting (non-trivial, implicit, previously unknown and potentially useful) patterns or knowledge from huge amount of data" as it is named in work [4].

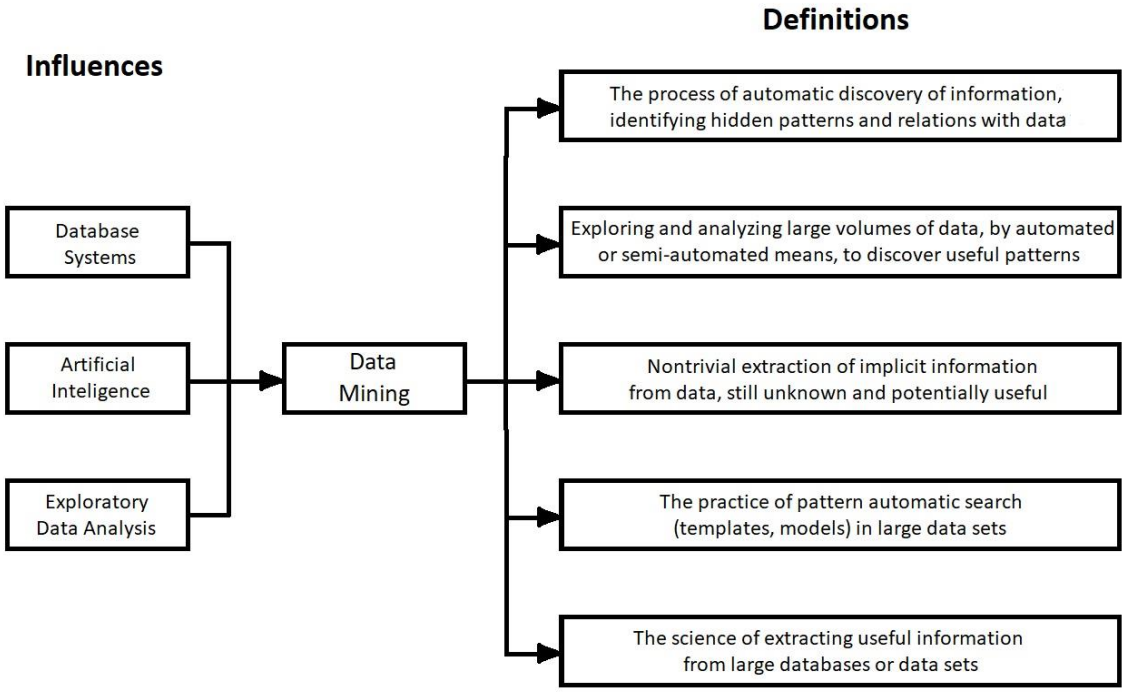


Figure 1 – Influences and definitions of data mining

A significant category of data mining techniques is that of mining frequent patterns, associations and correlations. Algorithms built for association rules are very useful from the perspective of marketing, because they develop methods for finding customers shopping patterns [6]. Applications of these special techniques are in basket data analysis, cross-marketing, catalogue design, sale campaign analysis, click stream of web logs analysis, and DNA sequence analysis [4]. In this paper we referred to that data mining component of association techniques and their analysis. We have created a remotely managed computer system, so that a Rapid Miner process, running through data mining operators and techniques, is written and uploaded to the application server and then accessed from our application, based on permanently updated data.

2 Process presentation

For determining the frequent item sets, we used the FP-Growth algorithm, which means Frequent Pattern Growth Algorithm, developed by J. Han, H. Pei, and Y. Yin [8]. This algorithm is a method for mining the complete set of frequent patterns by pattern fragment growth, using an extended prefix-tree structure for storing essential information about frequent patterns. This structure is known by the name of FP-tree. It is efficient, fast and scalable.
In the establishment of the association rules, we used the Apriori algorithm, written by Agrawal and Srikant in 1994 [9]. It determines the support of frequent sets of items by the method BFS (Breadth First Search). First, it determines the support of the sets one item, than with two items, continued recursively in the same way.
The RapidMiner environment contains a wide range of modular operators which allow the design of complex processing for a large number of data mining problems. An important characteristic of RapidMiner is the ability to imbricate operator chains and build trees of complex operators. We started from a process build through the work [5] but adapted it to a dataset stored in a MySQL database that can be stored on a server.

The dataset used in our process has the following structure of fields:

- *ID of movie* (a numerical value)
- *Name of movie* (nominal value)
- *ID of customer* (a numerical value)

Based on the dataset described above, we developed a data mining process using Rapid Miner, which will determine sets of frequent appearances from transactions, on which are generated association rules. Process built is shown in *Figure 2*.

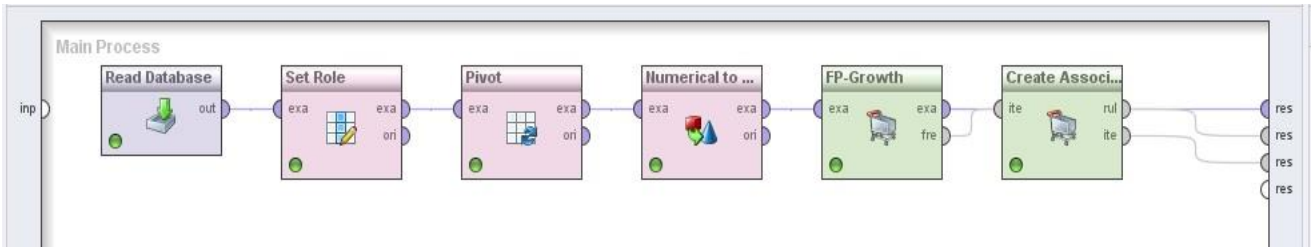


Figure 2 – The RapidMiner process for determining sets of frequent appearances and association rules generated with FP-Growth and Association Rules

For this process writing, created by facility GUI of Rapid Miner and refined programmatically through adequate XML code, for reasons of space, it will not be exposed in this article, we used the following operators:

- **Read Database** connects to a specified SQL database and reads an ExampleSet. In order to retrieve the data, a query can be specified. In our case, the query is:

```
SELECT `movies`.`movie_id` as MovieId, `transactions`.`customer_id` as CustomerId,
CONCAT(`movies`.` movie_id `, '_', `movies`.`name`) as MovieName
FROM `transactions`, `movies` WHERE `movie`.`movie_id` = `transactions`.`movie_id`
```

- **Set Role** is used by RapidMiner to change the role of one or more attributes. In our case we put value *ID of customer* to field ***attribute name***, ***target role*** received value *id*. Through option ***set additional roles*** we have established *Name of movie* as being *regular* type.
- **Pivot** is an important operator of this process and we used it to rotate the example set by grouping multiple examples of same groups to single examples. By option ***group attribute*** we selected the field *ID of customer*, by ***index attribute*** we have chosen the field *Name of Movie* and through ***weight aggregation*** we selected the option *count*.
- **Numerical to Binomial** changes the type of the selected numeric attributes to a binominal type. It is an essential operator from this process, because the operator with name ***FP-Growth*** works only with binomial values. We chose the option *all* for the option ***attribute filter type***.
- **FP-Growth** is a central operator of our construction. It calculates all frequent item sets from the given dataset using the *FP-tree* data structure. The range of values within which we chose *minimum support* for establishing frequent sets of items.
- **Create Association Rule** was written to obtain the association rules generated based on frequent occurrences of articles in transactions as they have been previous outcomes by using of operator, FP Growth. Data related to the values received by *minimum confidence* attribute, these constituting the support for the hypothesis of statistical analysis based on the results obtained.

3 Case study

In this part, we present how the Rapid Miner process mentioned in paragraph 2 can be applied into a real e-commerce application. Also, we explore how the library that realises the integration of Rapid Miner processes into Java applications can be used for a website created based on Java Server Pages technology.

3.1 The scenario

We have an online movie store. We need to make the most optimum associations, so the probability that the user that bought a movie would be interested in another suggested movie is as high as possible. The process presented in paragraph 2 in can be used to calculate which other movies are highly probable to be bought when one particular movie is bought, based on association rules, and also, what movies have the highest probability to be bought together, thus making possible to create and sell packages consisting of multiple movies.

3.2 The Application

The structure of the application is displayed in Figure 3.

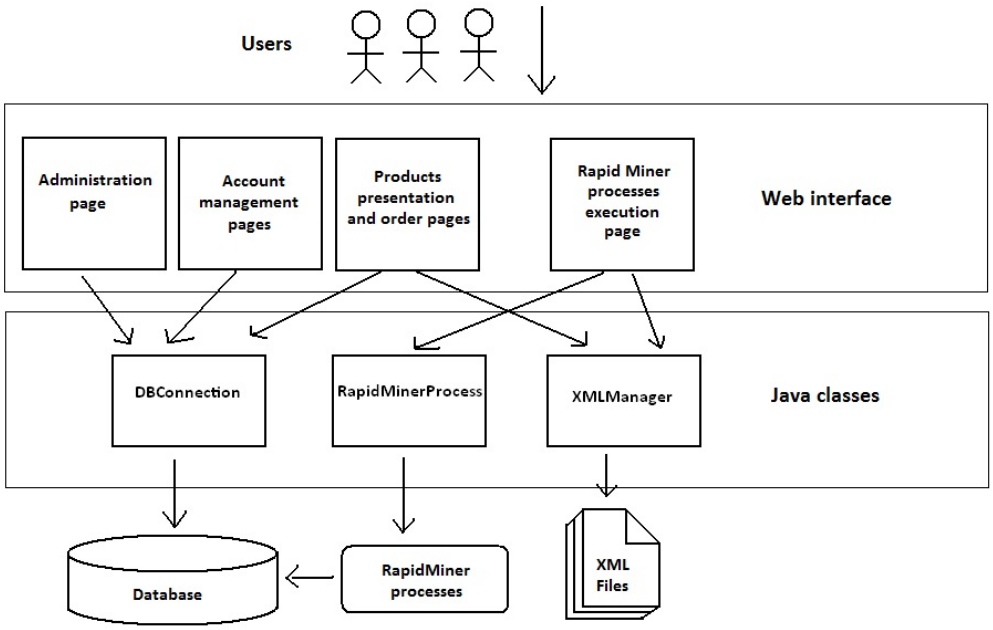


Figure 3 – Application structure

We want to point out the binding of the Rapid Miner process to the database that the application is build on, so every run of the process is using the most recent records. A particular Java class is used to change parameters and run the process, the methods being called by the users with administration rights on a special page.

3.3 Rapid Miner integration in Java class

Rapid Miner offers a package named com.rapidminer that can be included in Java projects. By the usage of this package, the user can import existing processes or breate new ones, change parameters, execute processes and display results. In this paragraph, we will present how we use in our application some important methods included in the package [7].

Initialize the Rapid Miner workspace:

```
RapidMiner.setExecutionMode(RapidMiner.ExecutionMode.COMMAND_LINE);
RapidMiner.init();
```

Import a process that is already created – Having the path of an existing process saved in a string variable named processRM, we create a new object of the Process class mapping the existing process by calling:

```
Process process= new Process(new File(processRM));
```

Retrieve an operator from a process – Having an object named process of type Process, to get an object of type Operator that represents the FP-Growth operator of the process, we call:

```
Operator operator = process.getOperator("FP-Growth");
```

Change a property of an operator – To set the minimum support of a FP-Growth operator, represented by an object named operator, to a value we have stored in a variable of type double named minSup, we execute the following line:

```
op.setParameter(FPGrowth.PARAMETER_MIN_SUPORT.Tools.formatNumber(minSup));
```

Run a process – The results of a process execution are stored in an object of IOContainer type. Having an object named process of class Process, the run method can be executed by calling:

```
IOContainer ioResult = process.run();
```

Parse process results – The result of a process execution, stored in an object of type IOContainer, are represented as a collection of objects implementing the IOObject interface. Each of the objects in the collection has a different type, depending on the operators present in the process. In our case, the second element will be an object of AssociationRules class, and the third element will be an object of FrequentItemSets class created by us, used to store the item name as string and the frequency as double.

- Parsing association rules:

```
IOObject result = ioResult.getElementAt(1);
List<String> associations = new ArrayList<String>();
if (result instanceof AssociationRules)
{
        String rules = result.toString();
        rules=rules.substring(rules.indexOf("\n")+1, rules.length());
        associations = Arrays.asList(rules.split("\r\n"));
}
```

- Parsing frequent item set:

```
IOObject result = ioResult.getElementAt(2);
List<FrequentItemSetClass> frequentItemSetDtos = new ArrayList<>();
if (result instanceof FrequentItemSets)
{
        for (FrequentItemSet fis : (FrequentItemSets) result)
        {
                FrequentItemSetClass frequentItemSetDto = new FrequentItemSetClass();
                frequentItemSetDto.setItem(fis.getItemsAsString());
                frequentItemSetDto.setFrequencyRatio(Tools.formatNumber((double) fis.getFrequency() /
                (double) ((FrequentItemSets) ioResult).getNumberOfTransactions()));
                frequentItemSetDtos.add(frequentItemSetDto);
        }
}
```

3.4 Interface for process execution

The section of the application used for executing Rapid Miner processes consists of two textboxes of numeric type, that can be used to change the minimum support and minimum confidence parameters, and three tables, that display the result of the process execution. In these tables we can see what movies are frequently bought together and what movies are most probably to be bought when one specific movie is bought. In Figure 4 we show the textboxes for the process inputs.

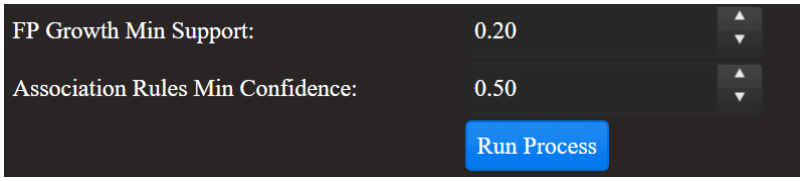


Figure 4 – Input fields for process parameters

The first input can be used to specify the mininum support parameter for the fp growth process, and the second numeric input can be used to specify the minimum confidence parameter for the assocation rules process.

Figure 5 – Process results and the way to display into application

In Figure 5 are presented together sections from the processes running page, movie information page that contains suggestions and packages, and home page which displays top movies of the moment.
In the left part of the figure is displayed a part of the detail page for the movie “Thor”. The middle part of the image contains two tables, representing the result of running the association rules process in the upper table and the result of running the fp growth process in the lower table. By looking in the upper table, we can see that the users who bought the movie “Thor” also bought „The Avengers and Iron Man, same things being suggested on the Thor detail page.
Also, by looking at the second table, we can see that the movie “Thor” was bought together with the movies “Iron Man”, “The Avengers” or both of them, so is more likely that other customers interested in the movie would buy a package containing these other movies.
In the right side of the figure, the upper part contains a section from the home page that displays the most popular movies since the last run of the processes. In the lower part, there is a table which display the movies that were bought most frequent, without being asociated with other movies, also coming as result from running the fp-growth process.

4 Conclusions

Data Mining Techniques for Associations Analysis is a modern and beneficial perspective in developing e-business activities, in particular the e-commerce component, through the marketing facilities offered. They enable both an activity of online marketing development through easy

access to products and the ability to make promotional packages as well as intelligent storage of products in warehouses. It is known that a smart, smart placement of products in a large deposit contributes to more efficient work, reducing time and spending on staff. In other words, the benefits are both on the customer side, through suggestions, promotional packs and time reduction, and on the vendor side, increasing employee productivity.
Integration of a Rapid Miner process into a Java application allows the user to change parameters as he wants. In this way, some parameters can be modified by someone who doesn't know how to use Rapid Miner, but knows what kind of result to expect after the process execution. Other advantage is that the system administrator can execute the process and check the result anytime and anywhere, using any device with Internet access and a browser installed, without the need to use the computing system where the database and Rapid Miner are installed.

References

[1] Christopher Clifton, *Encyclopedia Britannica: Definition of Data Mining*, https://www.britannica.com/technology/data-mining
[2] Aguiar-Pulido V., Seoane J.A., Gestal M., Dorado J., *Exploring patterns of epigenetic information with data mining techniques*, Curr Pharm Des. 2013; 19(4):779-89
[3] Kevin Roebuck, *Data Mining: High-Impact Strategies - What You Need to Know: Definitions, Adoptions, Impact, Benefits, Maturity, Vendors*, Lightning Source, 2011, ISBN 1743047029, 9781743047026
[4] Jiawei Han, Micheline Kamber, *Data Mining: Concepts and Techniques, 2nd ed.*, The Morgan Kaufmann Series in Data Management Systems, Jim Gray, Series Editor Morgan Kaufmann Publishers, March 2006. ISBN 1-55860-901-6, http://www.cs.uiuc.edu/homes/hanj/bk2/slidesindex.htm
[5] Mircea-Adrian Mușan, Ionela Maniu, *Extracting Associations Rules with FP-Growth and Apriori from Commercial Transactions*, "OVIDIUS" UNIVERSITY ANNALS – CONSTANTZA Year XVI – Issue 16 (2014) Series: CIVIL ENGINEERING
[6] Jerzy Korczak, Piotr Skrzypczak, *FP-Growth in Discovery of Customer Patterns*, Data-Driven Process Discovery and Analysis Lecture Notes in Business Information Processing Volume 116, 2012, pp 120-133, Print ISBN 978-3-642-34043-7, Online ISBN 978-3-642-34044-4
[7] RapidMiner classes documentation: http://fossies.org/linux/rapidminer/javadoc/overview-summary.html
[8] J. Han, H. Pei, and Y. Yin, *Mining Frequent Patterns without Candidate Generation*, In: Proc. Conf. on the Management of Data (SIGMOD'00, Dallas, TX). ACM Press, New York, NY, USA 2000
[9] Ch. Borgelt, *Frequent Pattern Mining*, Intelligent Data Analysis and Graphical Models Research Unit European Center for Soft Computing, 33600, Mieres, Spain, 2005

Mircea-Adrian MUȘAN
"Lucian Blaga" University of Sibiu
Department of Informatics
Sibiu, Street Ion Rațiu, No. 5
ROMANIA
E-mail: mircea.musan@ulbsibiu.ro

Iuliana-Maria CÂNDEA
"Lucian Blaga" University of Sibiu
Department of Informatics
Sibiu, Street Ion Rațiu, No. 5
ROMANIA
E-mail: iuliana.maria.candea@gmail.com

Fifth International Conference
Modelling and Development of Intelligent Systems
June 23 - 25, 2017
"Lucian Blaga" University
Sibiu - Romania

Theoretical and practical approaches for documents classification

Livia Sangeorzan, Nicoleta Enache-David

Abstract

Documents classification is a very actual issue and is a continuous challenge; it is based on different techniques of machine learning including Bayesian classification, SVM classifiers (Support Vector Machine), k-NN (k-Nearest-Neighbor) classifier, classification based on association rules, decision trees, etc.

In this paper we use Weka in order to make a comparison of the accuracy and error rates of two Bayesian classifiers, Naïve Bayes and Naïve Bayes Multinomial on a text training dataset.

1 Introduction

The task of classification occurs in a wide range of domains. The notion of *classification* could cover a context in which some decision is made on the basis of currently available information.

In [17] the authors consider that classification has two distinct meanings. Firstly, they consider a set of observations with the aim of establishing the existence of classes or clusters in the data. Secondly, they suppose they have a set of classes, and the scope is to establish a rule whereby they can classify a new observation into one of the existing classes. The first type is known as Unsupervised Learning and the second type as Supervised Learning.

Document classification is based on different techniques of machine learning, like Bayesian classification, Support Vector Machines [9],[10],[11].

Such methods can be applied on complex information systems like the ones presented in [1],[2],[3],[4] and for mathematical models like [6],[7],[8]. Also, in businesses, the decision making systems use decision trees [12],[13],[14].

In this paper we study the performance of two Bayesian classifiers: Naïve Bayes and Naïve Bayes Multinomial. We study also the amount of time taken to build the classification model.

2 Text classification approach

The problem of text classification consists of classifying documents by their content. Text classification is intended to assigning subjects to certain categories. Naive Bayes classifiers are create simple performing models, especially in the field of document classification. They are based on the Bayes' Theorem. [16]

There are several types of Naïve Bayes classifiers: Multinomial Naive Bayes, Binarized Multinomial Naive Bayes and Bernoulli Naive Bayes. Naïve Bayes and multinomial Naïve Bayes model are both supervised learning methods.

Each type of Naïve Bayes classifiers can have as output different results since they use completely different models.

In practice, it is possible to have more than two classes and the naïve Bayesian classifiers estimate the probability of class c_j generating instance d. Generally, the Naïve Bayes attributes have independent distributions. The assumption to have all attributes independent because of the meaning of the word naïve does not fit in real world situations.

We can give a definition for the text classification like the following: we have as input a document d, a fixed set of classes $C = \{c_1, c_2, ..., c_n\}$ and as output a predicted class $c \in C$ [17].

We denote by X the document space. In text classification, we are given a description $d \in X$ of a document and a fixed set of classes $C = \{c_1, c_2, ..., c_n\}$. Classes are called categories or labels.

3 Case study using Naïve Bayes and Naïve Bayes Multiomial classifiers

In our case study we use a text training dataset having 2132 words. We use Weka environment in order to study the accuracy and error rate when applying two bayesian classifiers: Naïve Bayes and Naïve Bayes Multinomial.

Weka is a collection of machine learning algorithms for solving real-world data mining problems. It is written in Java and runs on almost any platform. Features of Weka are: machine learning, data mining, preprocessing, classification, regression, clustering, association rules, attribute selection, visualization. [18]

In Weka we have chosen the Filtered Classifier from Meta category, and we made a comparison between Naïve Bayes and Naïve Bayes Multinomial classifiers implemented in Weka environment.

The dataset was tested using two methods for testing the accuracy: percentage split method, where 66% of the data was used as training dataset and 33% as testing dataset and the 10-fold cross validation method. We have obtained the results from the Table 1.

Method	**Accuracy**		**Error Rate**	
	Naïve Bayes	**Naïve Bayes Multinomial**	**Naïve Bayes**	**Naïve Bayes Multinomial**
Percentage Split 66%	81.81%	72.72%	18.18%	27.27%
10 Folds Cross Validation	71.87%	75%	28.12%	25%

Table 1. Accuracy and error rate for Naive Bayes and Naive Bayes Multinomial

From Figure 1 we can see that the Naïve Bayes classifier achieved the highest accuracy (81.81%) and the lowest error rate (18.18%) using the percentage split 66% option.

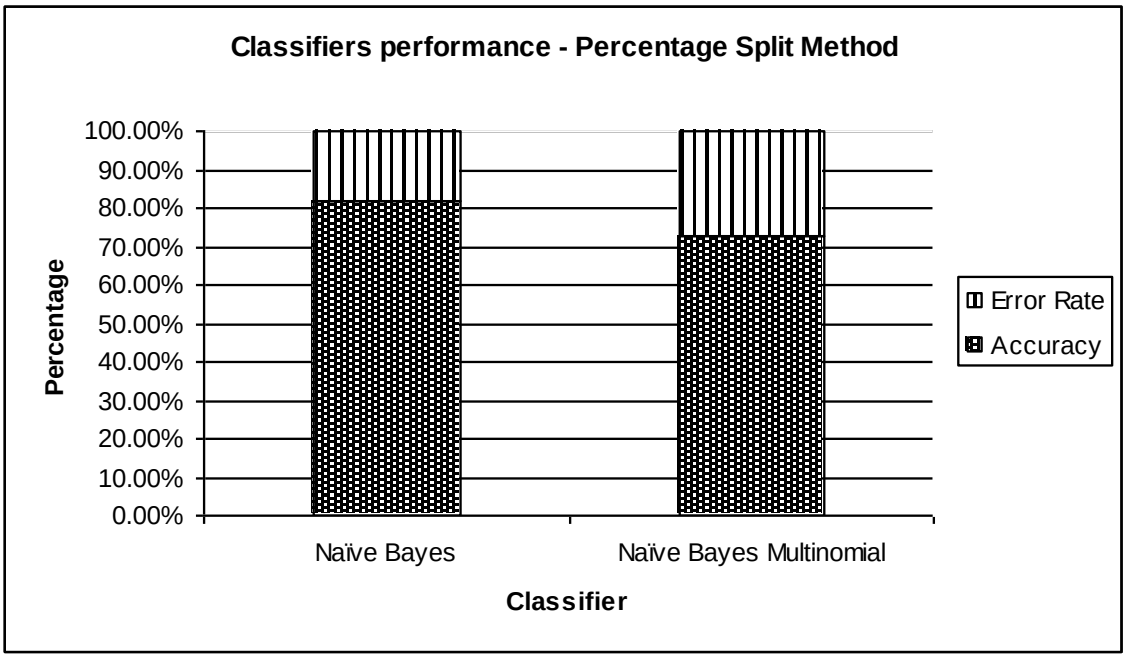


Fig. 1. Accuracy and Error Rate using percentage split method

From Figure 2 we can wee that Naïve Bayes Multinomial classifier achieved the highest accuracy (75%) and the lowest error rate (25%) using 10 folds cross validation method.

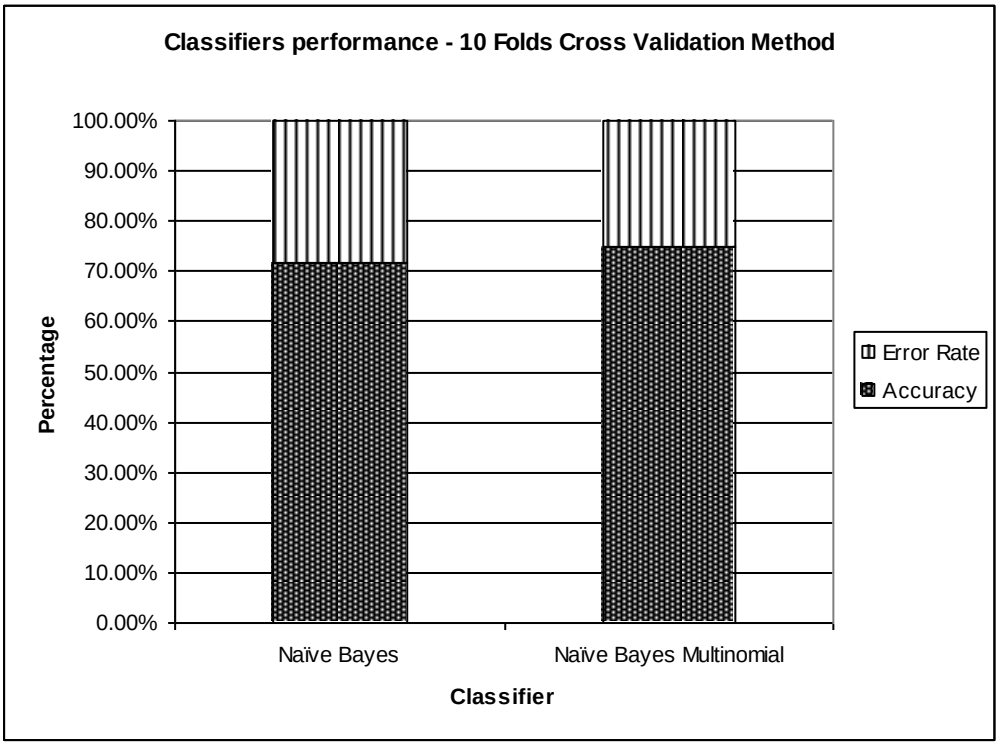


Fig. 2. Accuracy and Error Rate using 10 folds cross validation method

We have studied also the time needed for building the model for the two aforementioned classifiers. In table 2 we present the results.

Method	Time (sec)	
	Naïve Bayes	**Naïve Bayes Multinomial**
Percentage Split 66%	0.07	0.02
10 Folds Cross Validation	0.02	0.01

Table 2. Time to build the model for Naïve Bayes and Naïve Bayes Multinomial

The results show that the best amount of time was achieved by the Naïve Bayes Multinomial classifier with 10 folds cross validation method (0.01 sec), while Naïve Bayes classifier achieved 0.07 sec with the percentage split 66% method.

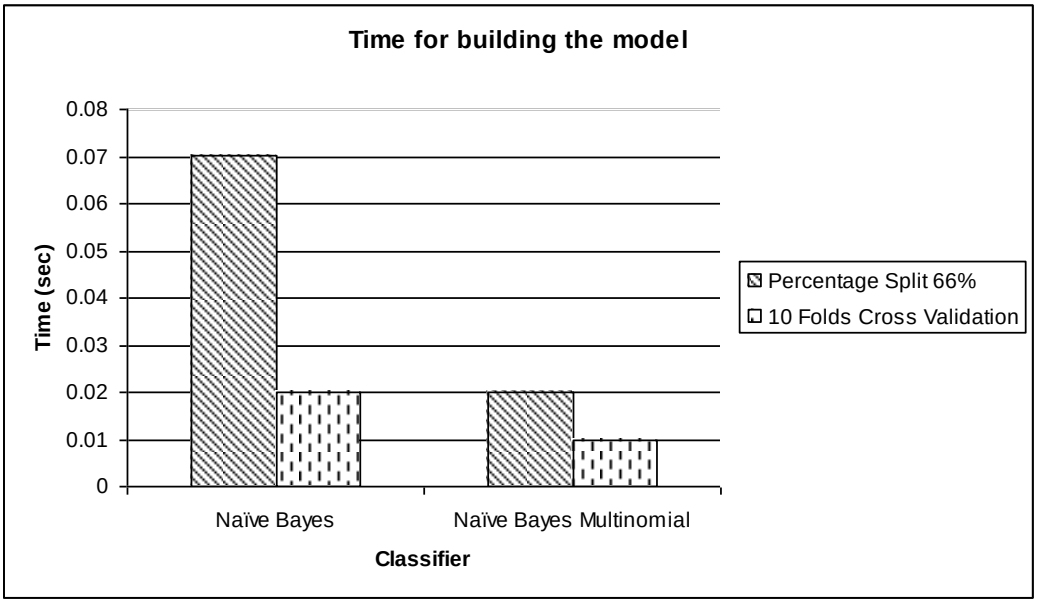


Fig. 3 Amount of time to build the model

4 Conclusion

In this paper we made a comparison regarding the performance of two types of Naïve Bayes classifiers on a document classification problem.

The conclusion is that the Naïve Bayes classifier achieved the highest accuracy using the percentage split 66% option, while the best amount of time was achieved by the Naïve Bayes Multinomial classifier with 10 folds cross validation method.

References

[1] C. Carstea, Control and management in complex information systems, *Bulletin of the Transilvania University of Braşov, Series III Mathematics • Informatics • Physics*, pp 73-87, 2013;

[2] C.G.Carstea, Modeling System's Process for Control Of Complex Information Systems, In Proceedings of The 25th International Business Information Management Association Conference (IBIMA), Soliman KS (ed). Amsterdam, Netherlands, May 2015, pp 566-574, 2015.
[3] C. Carstea, Optimization Techniques in Project Controlling, *OVIDIUS UNIVERSITY ANNALS, ECONOMIC SCIENCES SERIES*, Volume XIII Issue 1, pp. 428-432, Ovidius University Press, 2013.
[4] C.G. Carstea, IT Project Management – Cost, Time and Quality, ISSN 2067-5046, *Economy Transdisciplinary Cognition International* , Volume17, Issue 1/2014, pg.28-34, 2014.
[5] N. Enache-David, L. Sangeorzan, An overview on data mining techniques, *Proceedings of the 20th GBU-IC on Control, Development and APPLIED INFORMATICS in BUSINESS and ECONOMICS*, Brasov, Romania, November 1-2, Editura Didactica si Pedagogica Publishing House, Bucharest, Romania, ISSN:2069-7937, 2013.
[6] O. Florea, I. Rosca, The Mechanical Behavior and the Mathematical Modeling of an Intervertebral Disc, *Acta Technica Napocensis Series-Applied Mathematics Mechanics and Engineering*, 58: 213-218, 2015.
[7] O. Florea, I.C Rosca,, Analytic study of a rolling sphere on a rough surface, *AIP ADVANCES*, Volume: 6, Issue: 11, 2016.
[8] O. Florea, I.C. Rosca, Stokes' Second Problem for a Micropolar Fluid with Slip, *PLOS ONE*, Volume: 10, Issue: 7, Article Number: e0131860, 2015.
[9] A. Khashman, N.I. Nwulu, Support vector machines versus back propagation algorithm for oil price prediction, *Advances in Neural Networks*–ISNN 2011, pp 530-538, 2011.
[10] A. Khashman, Blood Cell Identification using Emotional Neural Networks, *J. Inf. Sci. Eng*. 25 (6), pp 1737-1751, 2009.
[11] A. Khashman, IBCIS: Intelligent blood cell identification system, *Progress in Natural Science 18* (10), PP1309-1314, 2008.
[12] L. Mandru, How to Control Risks? Towards A Structure of Enterprise Risk Management Process, *Journal of Public Administration, Finance and Law*, pp 80-92, 2016.
[13] M. Popescu, L. Mandru, Relationship between Quality Planning and Innovation, *Bulletin of the Transilvania University of Braşov*, Series V, Economic Sciences, Vol. 9 (58) No. 2, pp.203-212, 2016.
[14] L. Mandru, *Managementul integrat calitate-risc pentru societăţile comerciale cu profil industrial*, Editura Universităţii Transilvania, Brasov, 2011.
[15] C.D Manning, P. Raghavan, H. Schütze, *Introduction to Information Retrieval*, Cambridge University Press, 2008.
[16] H. Robbins, J. Ryzin, Introduction to statistics, Science research associates Inc., 1975.
[17] D. Michie, D.J. Spiegelhalter, C.C. Taylor, *Machine Learning, Neural and Statistical Classification*, Ellis Horwood Upper Saddle River, NJ, USA, 1994.
[18] http:// http://www.cs.waikato.ac.nz/ml/weka/.

LIVIA SANGEORZAN
Transilvania University
Department of Mathematics and Informatics
Str. Iuliu Maniu 50, Brasov
ROMANIA
E-mail: livia.sangeorzan@gmail.com

NICOLETA ENACHE-DAVID
Transilvania University
Department of Mathematics and Informatics
Str. Iuliu Maniu 50, Brasov
ROMANIA
E-mail: nicoleta.enache@unitbv.ro

Fifth International Conference
Modelling and Development of Intelligent Systems
June 23 - 25, 2017
"Lucian Blaga" University
Sibiu - Romania

Tuning Extreme Learning Machines with genetic algorithms

Florin Stoica, Alina Bărbulescu, Laura Florentina Stoica

Abstract

In this paper we propose a method of optimizing the predictions made with Extreme Learning Machines (ELM) by optimizing their structure. The method is based on generic algorithms to determine the optimal number of hidden nodes and also to determine the appropriate activation function. In our study the ELMs are optimized through the Breeder genetic algorithm aiming to minimize the prediction error for the sum of permanent premolars and canines dimensions from a group of 92 Romanian young people. A comparison is made with predictions obtained by multiple linear regression equations (MLRE).

1 Introduction

H.A.Boboc [4] developed multiple linear regression equations (MLRE) that investigates the relationship between the size of canine-premolar group and mesiodistal size of other teeth. The estimation of the mesiodistal size of permanent canine and of the two premolars before their eruption is important for early evaluation of the need for space in this area and represents an important part of diagnosis and orthodontic treatment strategy.
Our paper aims to verify if the Extreme Learning Machines with their structure optimized through genetic algorithms can improve the accuracy of the predictions provided by the original method based on MLRE.
A representative public school with a population of 321 children ages 12-15 years from Sibiu (Romania) was selected for this study. From these subjects, a random simple technique was used to select 92 students (47 females and 45 males) fulfilling the selection criteria:

- To have the parents' written consent to participate in the study;
- To present the dental arches fully erupted permanent teeth (molars 3 was not considered);
- The erupted teeth show no abnormalities of shape, size or structure;
- The teeth must not have missing of substance in the mesiodistal size, due to decay, trauma or orthodontic treatments have provided striping [8].

The measure tooth size models we used a digital calliper manufactured by Mega (Germany) with an accuracy of 0.01 mm. All models were measured 2 times by the same author and the result used was the average of two values.
For estimation the size of the unerupted canines and premolars, a recently multiple linear regression proposed equation [4] is based on known variables 21, 42 and 46. The form of this equation is: $Y = X_1 \times A_1 + X_2 \times A_2 + X_3 \times A_3 + A$, where:

- Y is the outcome expected;

- X1, X2, X3 are independent variables determined by the size of teeth 42, 46, 21;
- A1, A2 and A3 are regression coefficients for used teeth;
- A is a specific constant.

The values of constant A and regression coefficients of the equation are presented in Table 1:

Canines premolars group	Constant A	A_1 (42)	A_2 (46)	A_3 (21)
Maxillary	6,563	0,822	0,595	0,411
Mandible	3,350	0,872	0,710	0,538

Table 1: Parameters of multiple linear regression equation used [4]

In the following is presented our approach to provide a more accurate method for prediction of the mesiodistal width of unerupted permanent canines and premolars, using Extreme Learning Machines.

2 Extreme Learning Machines – fundamental concepts

Due to their remarkable efficiency, simplicity, and impressive generalization performance, ELMs have been applied in a variety of domains, such as control and robotics, computer vision, system identification, classification and regression.

The learning speed of feedforward neural networks is in general far slower than required and it has been a major bottleneck in their applications for past decades (the slow gradient-based learning algorithms are extensively used to train neural networks). Moreover, usually the parameters of the networks are tuned iteratively by using such learning slower algorithms.

Huang et al proposes a new learning algorithm called extreme learning machine (ELM) for single-hidden layer feedforward neural networks (SLFNs) [1].

In [1] is proved that the input weights and hidden layer biases of SLFNs can be randomly assigned if the activation functions in the hidden layer are infinitely differentiable. SLFNs can be simply considered as a linear system and the output weights (linking the hidden layer to the output layer) of SLFNs can be analytically determined through Moore–Penrose generalized inverse operation of the hidden layer output matrices.

For N arbitrary distinct samples (x_i, t_i), where $x_i = [x_{i1}, \ x_{i2}, \ . \ . \ ., \ x_{in}]^T \in \boldsymbol{R}^n$ and $t_i = [t_{i1}, \ t_{i2}, \ . \ . \ ., \ t_{im}]^T \in \boldsymbol{R}^m$ standard SLFNs with $\widetilde{\boldsymbol{N}}$ hidden nodes and activation function $g(x)$ are mathematically modeled as:

$$\sum_{i=1}^{\widetilde{N}} \beta_i \, g(w_i * x_j + b_i) = o_j$$

$j = 1, \ . \ . \ . \ , \boldsymbol{N}$ where:

- $w_i = [w_{i1}, \ w_{i2}, \ . \ . \ ., \ w_{in}]^T$ is the weight vector connecting the ith hidden node and the input nodes;
- $\beta_i = [\beta_{i1}, \ \beta_{i2}, \ . \ . \ ., \ \beta_{im}]^T$ is the weight vector connecting the ith hidden node and the output nodes;
- b_i is the threshold of the ith hidden node;

That standard SLFNs with $\widetilde{\boldsymbol{N}}$ hidden nodes with activation function $g(x)$ can approximate these N samples with zero error if $\sum_{j=1}^{N} \|o_j - t_j\| = 0$

In other words there exist β_i, w_i and b_i such that:

$$\sum_{i=1}^{\tilde{N}} \beta_i \, g\left(w_i * x_j + b_i\right) = t_j, \; j = 1, \ldots, N$$

The above N equations can be written as

$$\boldsymbol{H}\,\beta = \boldsymbol{T}$$

where $\boldsymbol{H}$ is called the ***hidden layer output matrix*** of the neural network [2] and can be described as follows:

$$\boldsymbol{H}\,(w_1, \ldots w_{\tilde{N}}, b_1, \ldots b_{\tilde{N}}, x_1, \ldots x_N) = \begin{bmatrix} g(w_1 * x_1 + b_1) & \ldots & g(w_{\tilde{N}} * x_1 + b_{\tilde{N}}) \\ \vdots & & \vdots \\ g(w_1 * x_N + b_1) & \ldots & g(w_{\tilde{N}} * x_N + b_{\tilde{N}}) \end{bmatrix}_{N \times \tilde{N}}$$

and β respectively T are matrices of form:

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_{\tilde{N}}^T \end{bmatrix}_{\tilde{N} \times m} \quad \text{and} \quad \boldsymbol{T} = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N \times m}$$

The ith column of H is the *i*th ***hidden node output*** with respect to inputs $x_1, \; x_2, \ldots, x_N$.
The input weights w_i and the hidden layer biases b_i are in fact not necessarily tuned and the hidden layer output matrix $\boldsymbol{H}$ can actually remain unchanged once random values have been assigned to these parameters in the beginning of learning.

For fixed input weights w_i and the hidden layer biases b_i, ***to train*** an SLFN is simply equivalent to ***finding a least squares solution*** $\dot{\beta}$ of the linear system $\boldsymbol{H}\,\beta = \boldsymbol{T}$.
In most cases the number of hidden nodes is much less than the number of distinct training samples $\tilde{N} \ll N$, $\boldsymbol{H}$ is a nonsquare matrix and there may not exist w_i, b_i, β_i such that $\boldsymbol{H}\,\beta = \boldsymbol{T}$
The smallest norm least squares solution of the above linear system is:

$$\dot{\beta} = \boldsymbol{H}^* \boldsymbol{T}$$

where H^* is the *Moore–Penrose* generalized inverse of matrix *H i.e.*

$HH^*H = H$
$H^*HH^* = H^*$
$(HH^*)^T = HH^*$
$(H^*H)^T = H^*H$

$\boldsymbol{H}^*$ can be calculate using the singular value decomposition (SVD) method [6].

3 Tuning ELMs with Breeder genetic algorithm

The Breeder genetic algorithm, proposed by Mühlenbein and Schlierkamp-Voosen [3] represents solutions (chromosomes) as vectors of ***real numbers***, much closer to the reality than normal GAs.
The ***selection*** is achieved randomly from the *T*% best elements of current population, where *T* is a constant of the algorithm (usually, *T* = 40 provide best results).

Within each generation, from the *T*% best chromosomes are selected two elements, and the ***crossover*** operator is applied over them.
On the new child obtained from the mate of the parents is applied the mutation operator. The process is repeated until are obtained *N-1* new individuals, where *N* represents the size of the initial population. The best chromosome (evaluated through fitness function) is inserted in the new population (1-elitism). Thus, the new population will have also *N* elements.

3.1 The Breeder genetic operators

3.1.1 Crossover

Let be $x=\{x_1, x_2, ..., x_n\}$ and $y=\{y_1, y_2, ..., y_n\}$ two chromosomes, where $x_i \in R$ and $y_i \in R, i=\overline{1,n}$. The crossover operator has a result a new chromosome, whose genes are represented by values $z_i = x_i + \alpha_i(y_i - x_i)$, $i=\overline{1,n}$, where α_i is a random variable uniformly distributed between $[-\delta, 1+\delta]$, and δ depends on the problem to be solved, typically in the interval $[0,0.5]$.

3.1.2 Mutation

The probability of mutation is typically selected as $1/n$. The mutation scheme is given by $x_i = x_i + s_i \cdot r_i \cdot a_i$, $i=\overline{1,n}$ where: $s_i \in \{-1,+1\}$ uniform at random, r_i is the range of variation for x_i, defined as $r_i = r \cdot domain_{x_i}$, where r is a value in the range between 0.1 and 0.5 (typically 0.1) and $domain_{x_i}$ is the domain of the variable x_i and $a_i = 2^{-k\cdot\alpha}$ where $\alpha \in [0,1]$ uniform at random and k is the number of bytes used to represent a number in the machine within is executed the Breeder algorithm (mutation precision).

3.2 The Breeder genetic algorithm

With the operators described in the previous section, the Breeder algorithm can be described as follows [7]:

Procedure Breeder
```
begin
t = 0
Randomly generate an initial population P(t) of N individuals
while (termination criterion not fulfilled) do
     Evaluate P(t) using the fitness function
     for i = 1 to N-1 do
         Randomly choose two elements from the T% best elements of P(t)
         Apply the crossover operator
         Apply the mutation operator on the child
         Insert the result in the new population P'(t)
     end for
     Choose the best element from P(t) and insert it into P'(t)
     P(t+1) = P'(t)
     t = t + 1
end while
end
```

3.3 Tuning parameters of ELM

The aim of the Breeder genetic algorithm is to find optimal values for the parameters of the ELM (the number of hidden nodes $\tilde{N}$ and the activation function g). Each chromosome contains two genes, representing values associated with modeled variables. The fitness function for chromosomes evaluation is represented by the train error of the represented ELM on train data sets.

In our tests, parameters of Breeder algorithm are assigned with following values: $\delta = 0$, $r = 0.1$, $a = 0.1$ and $k = 8$. The initial population has 100 chromosomes and algorithm is stopped after 100 generations.

Data provided by our study models was randomly divided in two sets: the training set, containing 50 cases and the validation set, composed by 42 study models (the data comes from 92 children ages 12-15 years fulfilling the selection criteria).

For the ELM, we chose between two activation functions, „*sig*" and „*sin*".

The original implementation of the elm-java tool can be found at http://www3.ntu.edu.sg/home/egbhuang/elm_codes.html [5].

3.4 Results

Using the data from training set, the Breeder algorithm has determined as optimal values for ELM parameters the "sig" as activation function and 15 as optimal value for the number of hidden nodes $\widetilde{\boldsymbol{N}}$.

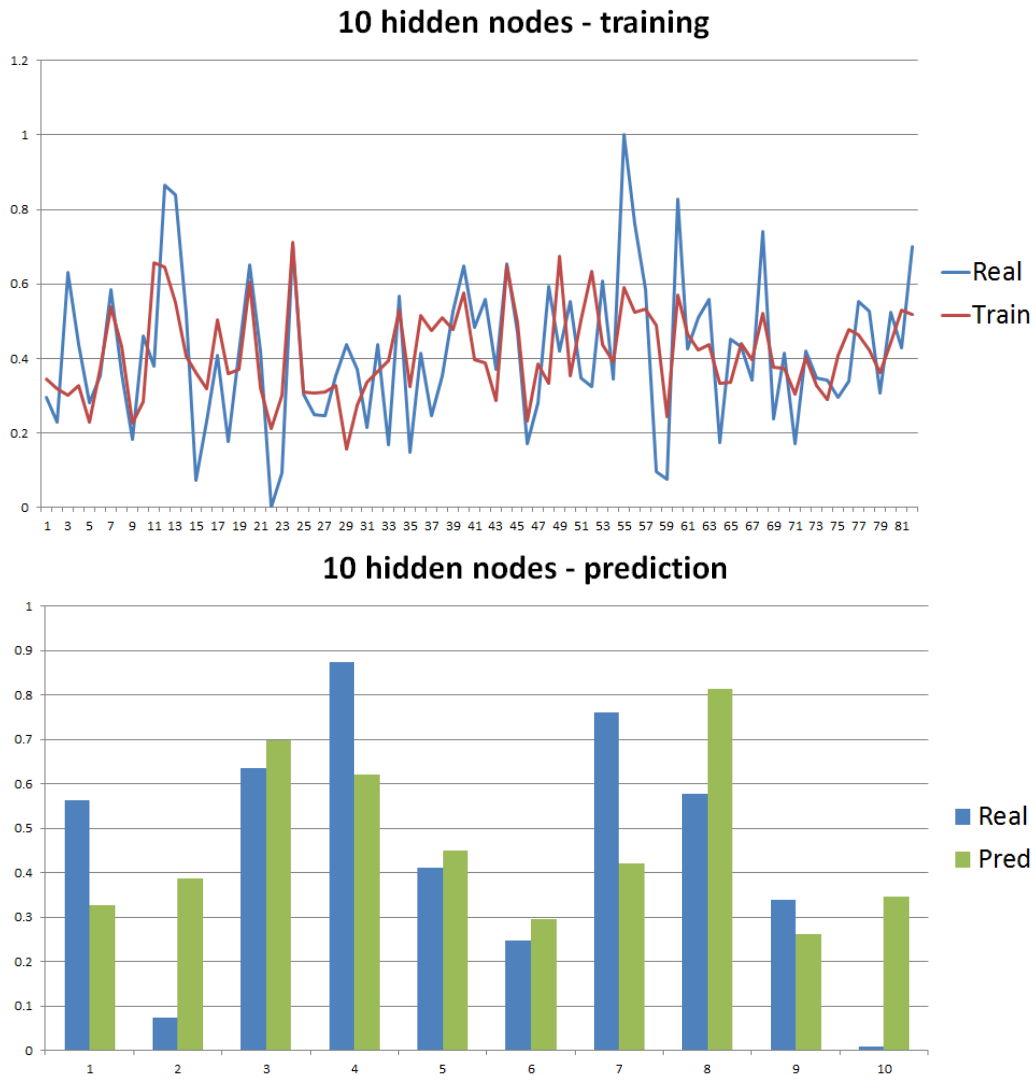


Figure 1: Training and prediction errors for ELM with $\widetilde{\boldsymbol{N}} = 10$

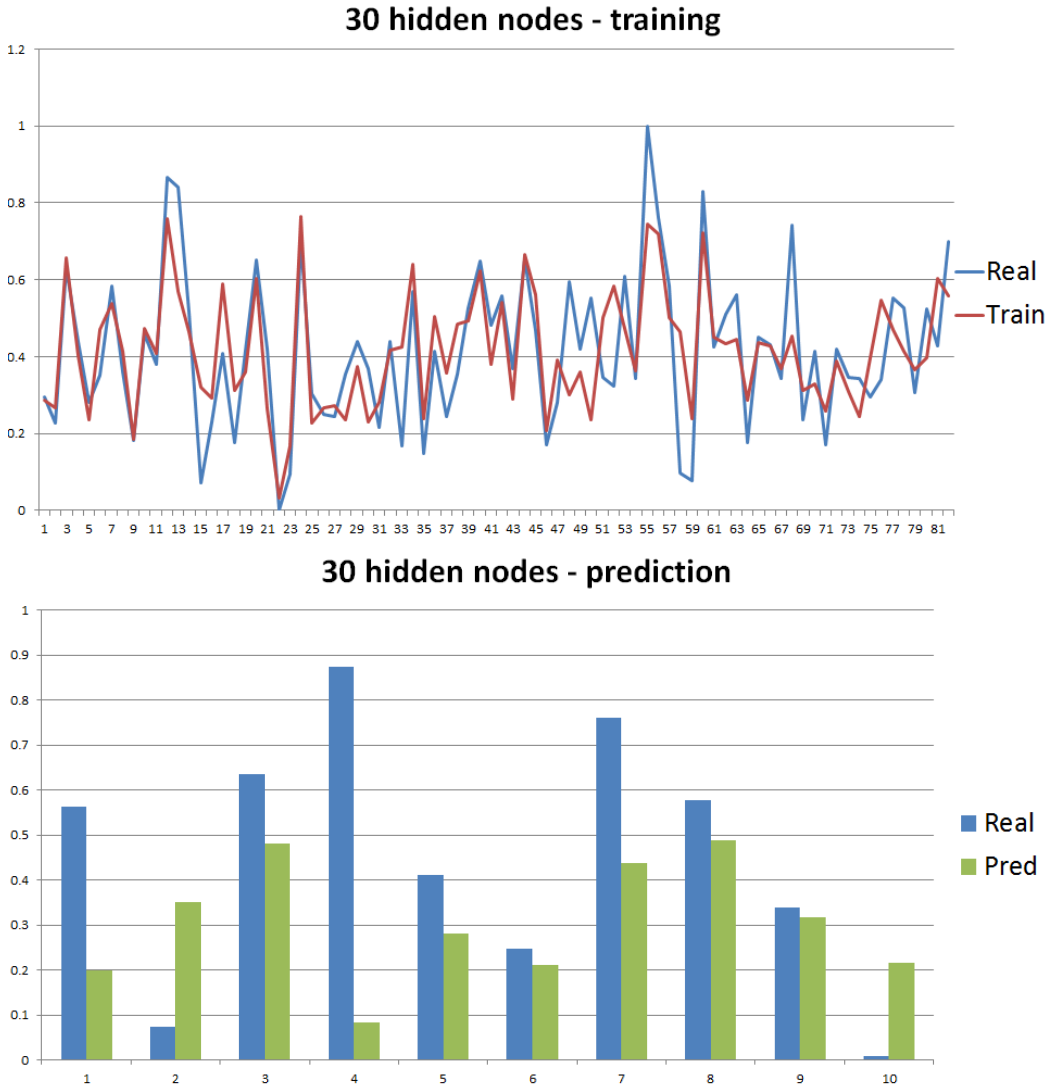
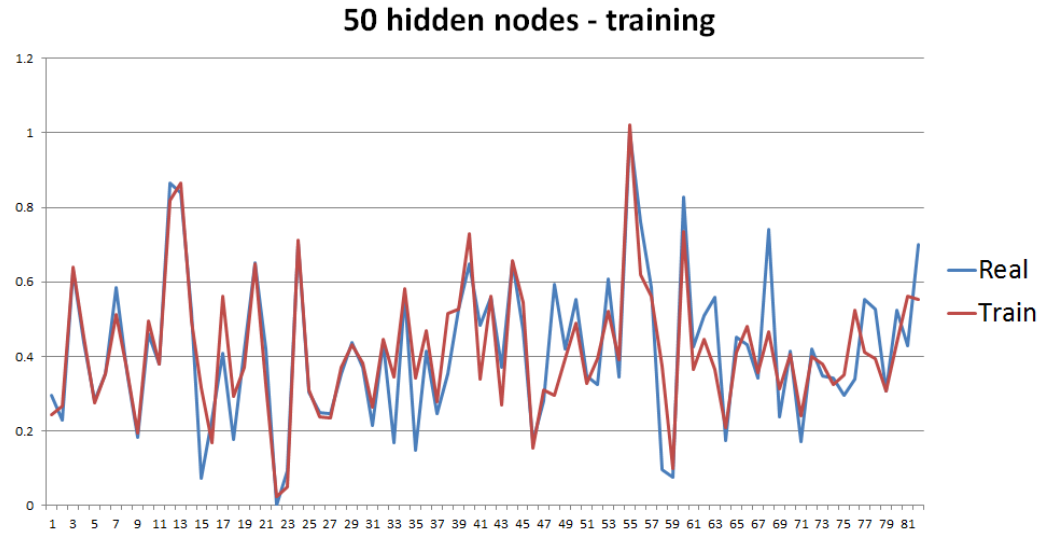


Figure 2: Training and prediction errors for ELM with $\tilde{N} = 30$



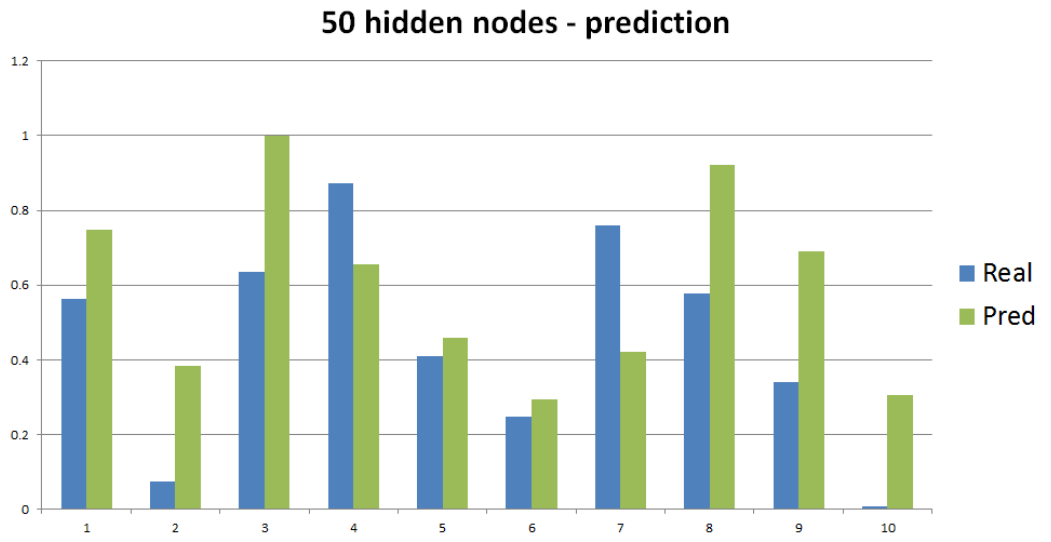


Figure 3: Training and prediction errors for ELM with $\widetilde{\boldsymbol{N}} = 50$

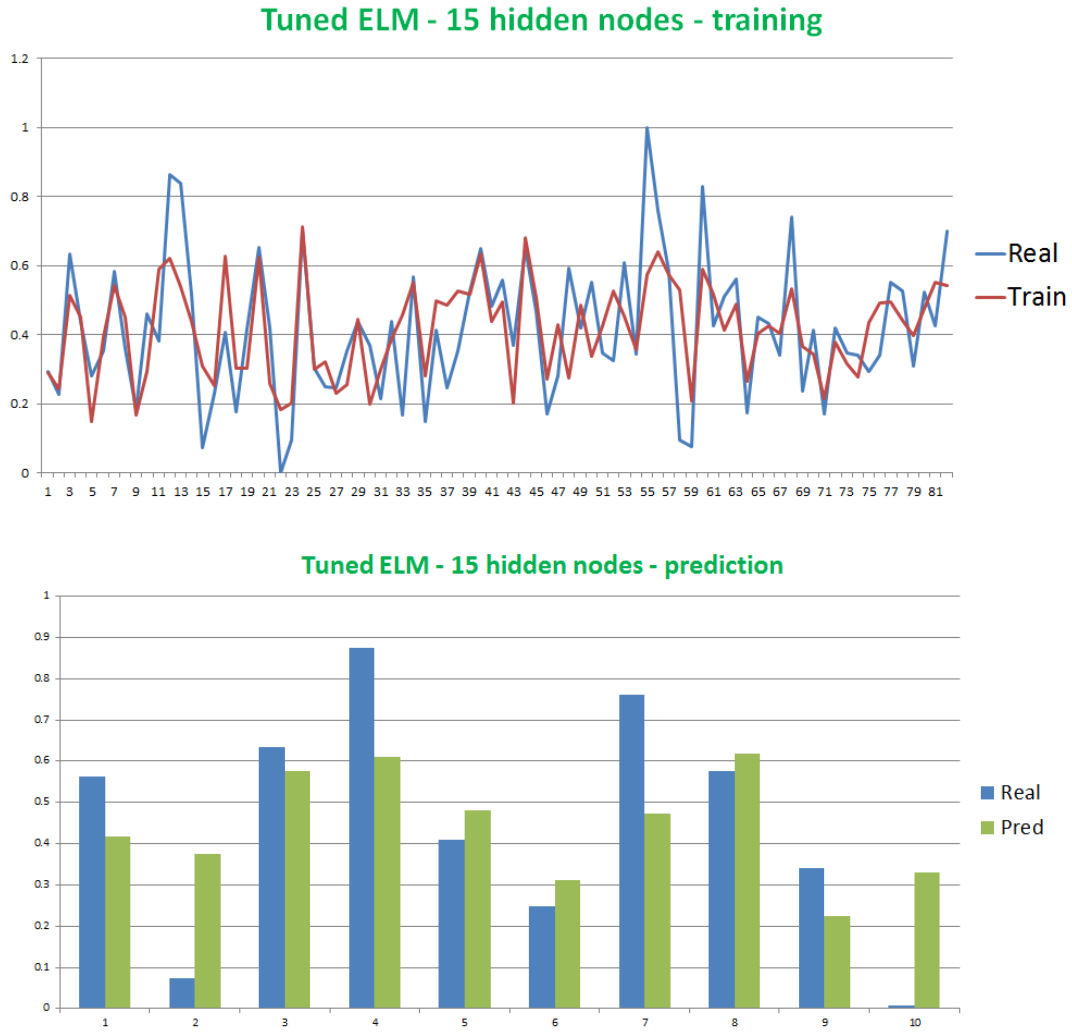


Figure 4: Training and prediction errors for ELM with $\widetilde{\boldsymbol{N}} = 15$

In order to evaluate the performance of the proposed approach, we carried out comparisons of results obtained by the tuned ELM with results provided by other ELMs with empirically established architecture. Also we made a comparison of the prediction accuracy between the two methods (tuned ELM vs. Boboc's multiple linear regression equations) using the following performance metrics and their formula:

- The metric $\boldsymbol{r}$ (Pearson's correlation coefficient) with formula: $\frac{\sum_{i=1}^{n}(p_i - m_p)(a_i - m_a)}{\sqrt{\sum_{i=1}^{n}(p_i - m_p)^2}\sqrt{\sum_{i=1}^{n}(a_i - m_a)^2}}$
- The metric RMSE (Root Mean Square Error) with formula: $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i - a_i)^2}$
- The metric MSE (Mean Squared Error) with formula: $\frac{1}{n}\sum_{i=1}^{n}(p_i - a_i)^2$
- The metric MAE (Mean Absolute Error) with formula: $\frac{1}{n}\sum_{i=1}^{n}|a_i - p_i|$

where notations are: p_i - predicted data; a_i - observed (actual) data; n - number of data; m_a - mean of observed data; m_p - mean of predicted data.

The comparison of performances of tuned vs. non-tuned ELMs are presented in Table 2.

	$\tilde{\boldsymbol{N}}$	10	**15**	30	50
Training	$\boldsymbol{r}$	0.6007745	0.6864645	0.7676291	0.8751079
	RMSE	0.1580064	0.143725	0.1266738	0.09564912
	MSE	0.02496602	0.02065688	0.01604626	0.009148755
	MAE	0.127673	0.1103865	0.09795375	0.06541024
Prediction	$\boldsymbol{r}$	0.5519677	**0.7211824**	0.09139431	0.5684539
	RMSE	0.2272806	**0.199109**	0.3211489	0.2759233
	MSE	0.05165646	**0.03964439**	0.1031366	0.07613369
	MAE	0.1943521	**0.1671411**	0.2393244	0.2504432

Table 2: Performance comparison with tuned ELM ($\tilde{\boldsymbol{N}} = 15$)

Comparing predictions provided by the proposed (ELM) and respectively MLRE method, we can conclude that the ELM tuned by a Breeder genetic algorithm is capable to provide greater accuracy in prediction of the mesiodistal width of unerupted teeth as can be seen in the figure 5 and respectively in table 3.

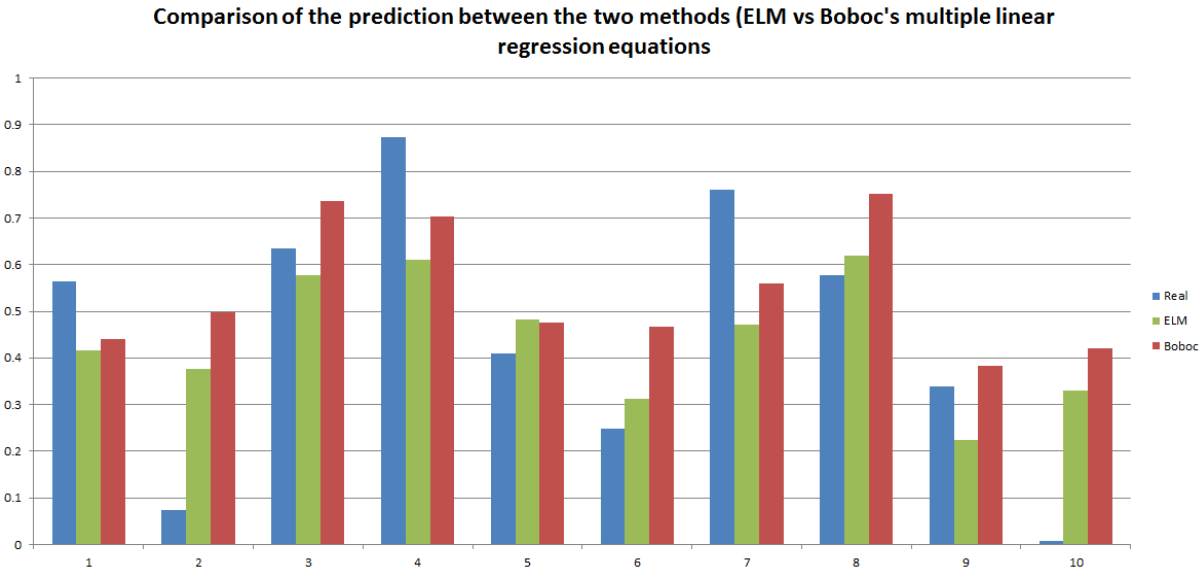


Figure 5: Comparing prediction errors for the two methods: ELM and MLRE

		ELM (15 nodes)	Boboc (MLRE)
Prediction	***r***	**0.7211824**	0.6519263
	RMSE	**0.199109**	0.230186
	MSE	**0.03964439**	0.05298558
	MAE	**0.1671411**	0.1938037

Table 3: Comparing prediction errors (ELM and MLRE) using performance metrics

4 Conclusions

Tuning the ELM using a genetic algorithm provide better results than empirical approach. The fast learning speed of an Extreme Learning Machine is crucial in obtaining a reasonable time for the tuning process. After evaluation, we found that our proposed method is providing a better prediction than original MLRE method. Thus, the prediction error rates of tuned ELM using the Breeder genetic algorithm are smaller than those provided by the multiple linear regression equations proposed in [4]. We intend to do more tests to be done on large data sets. Also, will be interesting to compare an ELM with a backpropagation trained neural network in terms of speed and prediction accuracy.

References

[1] G.-B. Huang, Q.-Y. Zhu, C.-K. *Siew Extreme learning machine: Theory and applications*, Neurocomputing *70* (2006) 489–501

[2] G.-B. Huang, *Learning capability and storage capacity of two hidden-layer feedforward networks*, IEEE Trans. Neural Networks, 14 (2) (2003) 274–281.

[3] Mühlenbein H, Schlierkamp-Voosen D, *The science of breeding and its application to the breeder genetic algorithm*, Evolutionary Computation, vol. 1, 1994, 335-360

[4] Boboc A, Dibbets J, *Prediction of the mesiodistal width of unerupted canines and premolars: a statistical approach*, *American Journal of Orthodontics and Dentofacial Ortopedics*, 2010, vol 137(4);503-507

[5] Dong Li, *Java version of ELM*, http://www3.ntu.edu.sg/home/egbhuang/elm_codes.html.

[6] J.M. Ortega, *Matrix Theory*, Plenum Press, New York, London, 1987.

[7] Stoica F., Simian D., *Optimizing a New Nonlinear Reinforcement Scheme with Breeder genetic algorithm*, Proceedings of the 11th International Conference on EVOLUTIONARY COMPUTING (EC'10),13-15 June 2010, Iaşi, Romania, ISSN: 1790-2769, ISBN: 978-960-474-194-6, pp. 273-278

[8] Florin Stoica, Cornel Gheorghe Boitor, *Using the Breeder genetic algorithm to optimize a multiple regression analysis model used in prediction of the mesiodistal width of unerupted teeth*, International Journal of Computers, Communications & Control, Vol 9, No 1, pp. 62-70, ISSN 1841-9836, february 2014, Impact Factor 2014 (JCR2013/Science Edition) 0.694

Florin STOICA
"Lucian Blaga" University of Sibiu
Faculty of Sciences
Mathematics and Informatics Department
Dr. Ion Rațiu St, 5-7 No, Sibiu, 550012
ROMANIA
E-mail: florin.stoica@ulbsibiu.ro

Alina BĂRBULESCU
Higher College of Technology
Sharjah
EMIRATELE ARABE UNITE
E-mail: alinadumitriu@yahoo.com

Florentina Laura STOICA
"Lucian Blaga" University of Sibiu
Faculty of Sciences
Mathematics and Informatics Department
Dr. Ion Rațiu St, 5-7 No, Sibiu, 550012
ROMANIA
E-mail: laura.cacovean@ulbsibiu.ro

Fifth International Conference
Modelling and Development of Intelligent Systems
June 23 - 25, 2017
"Lucian Blaga" University
Sibiu - Romania

Moth Search Algorithm for Bound Constrained Optimization Problems

Eva Tuba, Romana Capor-Hrosik, Adis Alihodzic, Marko Beko, Raka Jovanovic

Abstract

Numerous real life problems represent hard optimization problems that cannot be solved by deterministic algorithms. In the past decades, various different methods were proposed for these kind of problems and one of the methods are nature inspired algorithms, especially swarm intelligence algorithms. Moth search optimization algorithm (MSO) is on of the recent swarm intelligence algorithm that has not been thoroughly researched. In this paper we tested MSO algorithm on 15 standard benchmark functions and compared results with particle swarm optimization algorithm. Comparison show that MSO has good characteristics and it outperformed other approach from literature.

1 Introduction

Optimization represents an important field in applied mathematics. Most of the real world problems can be described as an optimization problem where the goal is to find minimum or maximum of the objective (fitness) function. The goal is to find the solution $x* \in A$ for the objective function $F : A \rightarrow R$ where $A \subset R^N$ that satisfy the following condition:

$$K(x) \leq K(x*), \tag{1}$$

in the case of the function maximization or, in the case of minimization:

$$K(x) \geq K(x*). \tag{2}$$

For solving simple optimization problems, various deterministic models were developed. The problem is that real world optimization problems are usually rather difficult and these deterministic methods cannot find the optimal solution or at least, they cannot find one in reasonable time. One well known example of hard optimization problem is traveling salesman which can be solved by heuristic (checking all possible solutions). The complexity of deterministic method is N! which will make calculation time unreasonably long even for rather small problem dimension.

For hard optimization problem, so-called NP difficult, it is easy to test the solution, to find the value of the fitness function, but it is hard to find the optimal one. In order to find the solution for these problems, stochastic algorithms have to be used. These algorithms are using random elements in their execution along with the set of rules for finding solution inside the domain of possible solutions. Due the random factors, stochastic algorithms produce different solutions in each run thus they have to be allowed to run

long enough, in order to give "good enough" each time. Common practice is to run stochastic algorithm several times and as final solution to use the average of the obtained results.

One large group of stochastic algorithms are based on imitation of natural phenomena. Empirically, it was shown that in this way, good results can be obtained, but it is not always completely understood how. Nowadays, nature inspired algorithms are common research topic and numerous methods were developed. They can be roughly categorized into three groups: evolutionary, artificial immune systems and swarm intelligence.

Evolution algorithms have been inspired by the evolutionary process, i.e. on the idea of the survival of fittest agent. Each generation of the solutions is made by combining the best solutions from previous generation or by mutating them while the worst solutions are not selected to survive. Mutation represents the random factor in this algorithms. By numerous iterations of breeding, where each generation is closer to the optimal solution, it can be expected that after enough iterations, "good enough" solution will be found. Well known algorithms in this group are genetic algorithm, differential evolution, genetic programming, etc.

Similar concept to the described one is used in artificial immune systems where negative selection is the main characteristic. These algorithms search for bad solutions and eliminate them from population which is the process inspired by natural immune systems in living beings.

Swarm intelligence algorithms are inspired by different phenomena that include numerous agents able to perform simple operations and communicate between them self. In these algorithms, movement of each organism is influenced by the information that it collected, by the information obtained from other agents and random factor. Algorithms can be inspired by animals in nature, music harmony, brain storm process, fireworks explosion, etc. Particle swarm optimization [1] and ant colony optimizations [2] are among the first swarm intelligence algorithms. Some other well known algorithms in this group are artificial bee colony algorithm, harmony search, bat algorithm, cuckoo search and many others.

In this paper we tested one of the recent swarm intelligence algorithm, moth search algorithm (MSA) proposed in 2016 by Gai Ge Wang. MSA is tested on benchmark functions CEC 2013 and compared to particle swarm optimization algorithm.

The rest of the paper is organized as follows. In Section 2, a short review of the swarm intelligence algorithms is given. Moth search algorithm is defined in Section 3. Experimental results are presented in Section 4 while the conclusion and future research are given in Section 5.

2 Swarm intelligence algorithms

Real world optimization problems are usually high dimensional and with rather difficult fitness function with large number of local extremes. In order to find the global optimum, optimization method needs to have a mechanize to escape from local optima. All optimization metaheuristics have two parts: exploration and exploitation, i.e. global and local search. Exploration ensures the search of the entire search space and it finds to promising areas. Around the found good solutions by the global search, i.e. promising areas, exploitation is used to try to find a better solutions.

Swarm algorithms use collaborative behavior of the simple individuals for solving the optimization problem. Swarms of ants and bees in their search for the path between home and food source, elephant population herding behavior and similar phenomena were the inspiration for these methods.

Artificial bee colony (ABC) imitates the foraging behavior of honey bee swarm. Exploitation and exploration processes were implemented by using three different types of bees: employed, onlookers and scouts. ABC algorithm have been shown to be rather effective for solving optimization problems [3]. Many upgraded and enhanced versions of ABC were proposed [4], [5] and also parallelized version in [6].

Cuckoo search (CS) represents swarm intelligence proposed by Yang and Deb in 2009 [7]. CS algorithm simulates the food search process by utilizing the Lévy flights. It was applied to numerous problems and it proved to be robust optimization method for the global optimization [8], image processing [9], engineering problems [10], etc.

Bat algorithm (BA) was also proposed by Yang in 2010 [11]. BA was inspired by the echolocation behavior of the bats. Algorithm parameters that controls exploration and exploitation are pulse rates of emission and loudness. Original and modified versions were used in various applications such as parameter tuning for support vector machine [12], handwritten digit recognition [13], etc.

Hybridization is rather common method for improving the quality of the swarm intelligence algorithm. For example, hybrid of ABC and firefly algorithm was proposed for portfolio optimization problem in [14] while in [15] it was used for problems with entropy constraint [16]. Seeker optimization algorithm (SOA) algorithm for global optimization was hybridized with ABC in [17] and in [18] hybrid between SOA and firefly algorithm was proposed.

3 Moth Search Optimization Algorithm

Moth search (MS) algorithm is recent member of the swarm intelligence algorithms. It was proposed by Wang in 2016 [19]. Inspiration for the MS algorithm was the phototaxis and Lévy flights of the moths. It was tested on numerous benchmark function such as IEEE CEC 2005 and IEEE CEC 2011 and compared to five metaheuristic algorithms. Based on the results presented in [19], it can be concluded that it has great potential for solving optimization problems. In this paper MS was applied to the newer benchmark function, IEEE CEC 2013, in order to test the quality of the algorithm for the more complex tasks.

As it was mentioned, MS algorithm was inspired by the moth behavior in the nature where phototaxis and Lévy flights represent the most significant characteristics of moths which was used for implementing an optimization algorithm.

Phototaxis represent the moths tendency to fly around the light source. They will fly in a straight line in order to stay at a fixed angle to the celestial light [19]. Lévy flight is one of the most important flight patterns in natural surroundings. The form of Lévy flights can be approximated by the power law distribution over a range of scales with the feature of exponents close to 3/2. Lévy flights was introduced to other optimization algorithms such as cuckoo search [7], firefly algorithm [20], krill herd optimization algorithm [21], bat algorithm [22], etc.

Lévy flights represent type of random walk where the step length is drawn from Lévy distribution modeled in the form of a power-law formula [19]:

$$L(s) \sim |s|^{-\beta}, \tag{3}$$

where $\beta \in [0, 3]$ is an index.

In [23], movements with $\beta \approx 1.5$ for Lévy flights were used which was also accepted in this paper.

Phototaxis and Lévy flights from moths in nature were used for implementing exploitation and exploration.

The best solution in the population is considered as the light source. The solutions (moths) closer to the best one will fly around the best moth it in the form of Lévy flights. New solutions, i.e. movements of the previous, are defined by the following equation [19]:

$$x_i^{t+1} = x_i^t + \alpha L(s), \tag{4}$$

where x_i^{t+1} and x_i^t are i^{th} solutions in the generation $t + 1$ and t, respectively. Step size based on the Lévy distribution is $L(s)$. Parameter α represents the scale factor and its value depends on the considered optimization problem. In [19], α was determined by:

$$\alpha = S_{max}/t^2, \tag{5}$$

where S_{max} represents the maximum walk step.

Lévy distribution in Eq. (4) is calculated by the following equation [19]:

$$L(s) = \frac{(\beta-1)\Gamma(\beta-1)\sin(\frac{\pi(\beta-1)}{2})}{\pi s^{\beta}}, \tag{6}$$

where Γ stands for the gamma function and $s > 0$.

Solutions that are on large distance from the best solution in the current generation fly towards the light source in line. These solutions are generated by the following formula [19]:

$$x_i^{t+1} = \lambda \times (x_i^t + \phi \times (x_{best}^t - x_i^t)), \tag{7}$$

where x_{best}^t is the best solution in generation t while ϕ and λ represent acceleration and scale factors, respectively.

Some solutions are defined by moving them beyond the best solutions in the current generation. These new solutions are generated by the following equation [19]:

$$x_i^{t+1} = \lambda \times (x_i^t + \frac{1}{\phi} \times (x_{best}^t - x_i^t)) \tag{8}$$

Initial population is randomly generated. In each iteration, new set of solutions is calculated by updating the previous solution by using Eq. (4), Eq. (7) or Eq. (8). In this paper we used the same simplified process as in the original paper [19]. Whole population was divided into two equal groups based on their fitness function values. In the first group, new solutions are obtained by the Lévy flights (Eq. (4)). The second group where are the solutions with worse fitness function values, update their solutions b Eq. (7) or Eq. (8) with possibility of $p = 0.5$ [19].

4 Experimental Results

Moth search algorithm was implemented by using Matlab R2016a and experiments were conducted on the platform with Intel ® Core™ i7-3770K CPU at 4GHz, 8GB RAM, Windows 10 Professional OS.

MS algorithm was tested on fifteen standard benchmark functions proposed for CEC 2013 competition [24]. Used function along with their optimal fitness function values are listed in Table 1. Five unimodal and ten basic modal functions were used.

MS algorithm (MSA) was compared to the results obtained by particle swarm optimization (PSO) reported in [25]. For each test function algorithm was run 30 times. In Table 2 the best, the worst, median and standard deviation of the results obtained in 30 runs are presented.

As it can be seen, both, PSO and MSA, found the optimal function value for f_1 (sphere). Standard deviation was 0 which means that the optimal solution was found in each run by both algorithms. MSA algorithm found the optimal value for f_5 (different powers function) also with standard deviation 0 while PSO found the optimal value on the presented accuracy but standard deviation was greater then 0, which means that it does not find exactly the optimal solutions. Both algorithms were unable to find even close solutions to the optimal ones for functions f_2, f_3 and f_4. These function are specific and special parameter tuning is necessary and probably more iterations. Even in such conditions, MSA was able to find significantly better solutions compared to the PSO. Similar situation was for the functions f_{14} and f_{15} where even though MSA found rather larger fitness function values compared to the optimal, they were still radically better then the solutions obtained by PSO.

Table 1: Benchmark function details

No	Function	Optimal
Unimodal functions		
1	Sphere function	-1400
2	Rotated high conditioned elliptic function	-1300
3	Rotated bent cigar function	-1200
4	Rotated discus function	-1100
5	Different powers function	-1000
Basic multimodal functions		
6	Rotated Rosenbrock's function	-900
7	Rotated Schaffers F7 function	-800
8	Rotated Ackley's function	-700
9	Rotated Weierstrass function	-600
10	Rotated Griewank's function	-500
11	Rastrigin's function	-400
12	Rotated Rastrigin's function	-300
13	Non-Continuous rotated Rastrigin's function	-200
14	Schwefel's Function	-100
15	Rotated Schwefel's Function	100

MS algorithm obtained better mean and the best fitness function values for f_6 whit lower standard deviation which means that MSA is more stable then the PSO for this function. The worst solution were the same for the both algorithms.

For the functions f_7 to f_{13}, MSA algorithm found better median solutions and in the most cases the best and the worst solutions were better (if not better then the same) compared to the fitness function values obtained by the PSO. Standard deviation was smaller in all cases, except for f_8 and f_{12}. Smaller standard deviation around the worse solutions does not represent an advantage.

The quality of the MSA for functions f_2, f_3,f_4, f_{14} and f_{15} can be probably increased by adjusting the parameters additionally and/or by increasing the maximal iteration numbers.

Based on the result analysis, it can be concluded that MSA has good qualities for solving hard optimization problems and it obtain better results compared to the standard particle swarm optimization.

5 Conclusion

In this paper, recent swarm intelligence algorithm, moth search algorithm was tested on CEC 2013 benchmark functions for unconstrained single objective optimization problems. MSA was compared to the standard particle swarm optimization algorithm and it obtained better results for all test functions. Moreover, MSA found rather good solutions which proves the quality. In further work, MS algorithm

Table 2: Comparison of PSO and MSA

Function		**PSO**	**MSA**
f_1	median	-1.400E+03	-1.400E+03
	std	0.000E+00	0.000E+00
	best	-1.40E+03	-1.400E+03
	worst	-1.40E+03	-1.400E+03
f_2	median	3.50E+04	**2.934E+04**
	std	7.36E+04	**8.328E+04**
	best	7.597E+02	**1.853E+02**
	worst	4.755E+05	**4.129E+05**
f_3	median	2.67E+05	**1.284E+05**
	std	1.66E+07	**6.834E+06**
	best	**-1.200E+03**	**-1.158E+03**
	worst	8.251E+07	**1.795E+08**
f_4	median	7.769E+03	**2.359E+03**
	std	4.556E+03	**1.631E+03**
	best	2.454E+02	**1.195E+02**
	worst	1.856E+04	**5.270E+03**
f_5	median	-1.000E+03	-1.000E+03
	std	3.142E-05	**0.000E+00**
	best	-1.000E+03	-1.000E+03
	worst	-1.000E+03	-1.000E+03
f_6	median	-8.902E+02	**-8.256E+02**
	std	4.974E+00	**3.638E+00**
	best	-9.000E+02	**-9.000E+02**
	worst	-8.898E+02	-8.898E+02
f_7	median	-7.789E+02	**-7.582E+02**
	std	1.327E+01	**1.170E+01**
	best	-7.974E+02	**-7.697E+02**
	worst	-7.434E+02	**-7.382E+02**
f_8	median	-6.789E+02	**-6.797E+02**
	std	**6.722E-02**	4.338E-03
	best	-6.789E+02	**-6.797E+02**
	worst	-6.796E+02	**-6.797E+02**
f_9	median	-5.952E+02	**-5.969E+02**
	std	1.499E+00	**1.039E+00**
	best	-5.987E+02	**-5.991E+02**
	worst	-5.929E+02	-5.929E+02
f_{10}	median	-4.999E+02	**-4.999E+02**
	std	2.713E-01	**1.449E-01**
	best	-4.999E+02	**-5.000E+02**
	worst	-4.989E+02	**-4.984E+02**
f_{11}	median	-3.891E+02	**-3.907E+02**
	std	5.658E+00	**4.198E+00**
	best	-3.970E+02	**-3.972E+02**
	worst	-3.731E+02	**-3.781E+02**
f_{12}	median	-2.861E+02	**-2.870E+02**
	std	**6.560E+00**	6.019E+01
	best	-2.970E+02	**-2.971E+02**
	worst	-2.682E+02	-2.623E+02
f_{13}	median	-1.792E+02	**-1.801E+02**
	std	9.822E+00	**8.992E+00**
	best	-1.946E+02	**-1.992E+02**
	worst	-1.523E+022	**-1.617E+02**
f_{14}	median	7.338E+02	**2.914E+02**
	std	1.282E+02	1.282E+02
	best	2.228E+02	**-1.419E+02**
	worst	1.109E+03	**4.990E+02**
f_{15}	median	8.743E+02	**5.695E+02**
	std	2.507E+02	**2.429E+02**
	best	4.372E+02	**4.271E+02**
	worst	1.705E+03	**1.044E+03**

can be applied to some real world optimization problems. Also, it can be adjusted for solving constrained of multi-objective problem or improved by some hybridization, chaotic maps, etc.

References

[1] C., E.R., Kennedy, J.: A new optimizer using particle swarm theory. In: Proceedings of the 6th international symposium on micromachine and human science, IEEE (1995) 39–43

[2] Marco, D., M., G.L.: Ant colonies for the travelling salesman problem. Biosystems **43** (1997) 73–81

[3] Darvis, Akay, B.: A modified artificial bee colony (ABC) algorithm for constrained optimization problems. Applied Soft Computing **11** (2011) 3021–3031

[4] Brajevic, I., Tuba, M.: An upgraded artificial bee colony algorithm (ABC) for constrained optimization problems. Journal of Intelligent Manufacturing **24** (2013) 729–740

[5] Tuba, M., Bacanin, N., Stanarevic, N.: Adjusted artificial bee colony (ABC) algorithm for engineering problems. WSEAS Transactions on Computers **11** (2012) 111–120

[6] Subotic, M., Tuba, M.: Parallelized multiple swarm artificial bee colony algorithm (MS-ABC) for global optimization. Studies in Informatics and Control **23** (2014) 117–126

[7] Yang, X.S., Deb, S.: Cuckoo search via levy flights. In: Proceedings of World Congress on Nature & Biologically Inspired Computing (NaBIC 2009). (2009) 210–214

[8] Bacanin, N.: Implementation and performance of an object-oriented software system for cuckoo search algorithm. International Journal of Mathematics and Computers in Simulation **6** (2010) 185–193

[9] Brajevic, I., Tuba, M.: Cuckoo search and firefly algorithm applied to multilevel image thresholding. In Yang, X.S., ed.: Cuckoo Search and Firefly Algorithm: Theory and Applications. Volume 516 of Studies in Computational Intelligence., Springer International Publishing (2014) 115–139

[10] Yang, X.S., Deb, S.: Engineering optimization by cuckoo search. International Journal of Mathematical Modeling and Numerical Optimization **1** (2010) 330–343

[11] Yang, X.S.: A new metaheurisitic bat-inspired algorithm. Studies in Computational Intelligence **284** (2010) 65–74

[12] Tuba, E., Tuba, M., Simian, D.: Adjusted bat algorithm for tuning of support vector machine parameters. In: Congress on Evolutionary Computation (CEC),, IEEE (2016) 2225–2232

[13] Tuba, E., Tuba, M., Simian, D.: Handwritten digit recognition by support vector machine optimized by bat algorithm. In: 24th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision,(WSCG 2016). (2016) 369–376

[14] Tuba, M., Bacanin, N.: Artificial bee colony algorithm hybridized with firefly metaheuristic for cardinality constrained mean-variance portfolio problem. Applied Mathematics & Information Sciences **8** (2014) 2831–2844

[15] Bacanin, N., Tuba, M.: Firefly algorithm for cardinality constrained mean-variance portfolio optimization problem with entropy diversity constraint. The Scientific World Journal, special issue Computational Intelligence and Metaheuristic Algorithms with Applications, **2014** (2014) 16

[16] Tuba, M.: Asymptotic behavior of the maximum entropy routing in computer networks. Entropy **15** (2013) 361–371

[17] Tuba, M., Brajevic, I., Jovanovic, R.: Hybrid seeker optimization algorithm for global optimization. Applied Mathematics and Information Sciences **7** (2013) 867–875

[18] Tuba, M., Bacanin, N.: Improved seeker optimization algorithm hybridized with firefly algorithm for constrained optimization problems. Neurocomputing **143** (2014) 197–207

[19] Wang, G.G.: Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems. Memetic Computing (2016)

[20] Yang, X.S.: Firefly algorithms for multimodal optimization. Stochastic Algorithms: Foundations and Applications, LNCS **5792** (2009) 169–178

[21] Gandomi, A.H., Alavi, A.H.: Krill herd: A new bio-inspired optimization algorithm. Commun Nonlinear Sci Numer Simulat **17** (2012) 4831–4845

[22] Lin, J.H., Chou, C.W., Yang, C.H., Tsai, H.L., et al.: A chaotic levy flight bat algorithm for parameter estimation in nonlinear dynamic biological systems. Computer and Information Technology **2** (2012) 56–63

[23] Reynolds, A., Reynolds, D., Smith, A., Svensson, G., Lfstedt, C.: Appetitive flight patterns of male agrotis segetum moths over landscape scales. Journal of Theoretical Biology **245** (2007) 141 – 149

[24] Liang, J., Qu, B., Suganthan, P., 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)

[25] Zambrano-Bigiarini, M., Clerc, M., Rojas, R.: Standard particle swarm optimisation 2011 at CEC-2013: A baseline for future PSO improvements. In: IEEE Congress on Evolutionary Computation (CEC 2013), IEEE (2013) 2337–2344

Eva Tuba
John Naisbitt University
Graduate School of Computer Science
Bulevar umetnosti 29
SERBIA
E-mail: *etuba@ieee.org*

Romana Capor-Hrosik
University of Dubrovnik
Institute for Marine and Coastal Research
Kneza Damjana Jude 12
CROATIA
E-mail: *rcapor@undu.hr*

Adis Alihodzic
University of Sarajevo
Faculty of Mathematics
Zmaj od Bosne 33-35
BOSNIA AND HERZEGOVINA
E-mail: *adis.alihodzic@pmf.unsa.ba*

Marko Beko
Universidade Lusofona de Humanidades e Tecnologias
Campo Grande 376, Lisbon
PORTUGAL
E-mail: *marko@isr.ist.utl.pt*

Raka Jovanovic
Hamad Bin Khalifa University
Doha
QATAR
E-mail: *rjovanovic@qf.org.qa*

Author Index

Alihodzic, Adis	*University of Sarajevo, Bosnia and Herzegovina*	82
Alina, Bărbulescu	*Higher College of Technology, Sharjah, UAE*	72
Bănăduc, Angela	*"Lucian Blaga" University of Sibiu, Romania*	43
Bănăduc, Doru	*"Lucian Blaga" University of Sibiu, Romania*	43
Beko, Marko	*Universidade Lusofona de Humanidades e Tecnologias, Lisbon, Portugal*	82
Bota, Petrică	*"Lucian Blaga" University of Sibiu, Romania*	13
Cândea, Iuliana Maria	*"Lucian Blaga" University of Sibiu, Romania*	60
Capor-Hrosik, Romana	*University of Dubrovnik, Croatia*	82
Ciurea, Stelian	*"Lucian Blaga" University of Sibiu, Romania*	20
Constantinescu, Zoran	*Petroleum-Gas University of Ploiești, Romania*	50
Enache-David, Nicoleta	*Transilvania University of Brașov, Romania*	67
Holmstedt, Viggo	*University College of Southeast, Norway*	29
Hunyadi, Daniel	*"Lucian Blaga" University of Sibiu, Romania*	43
Jovanovic, Raka	*Hamad Bin Khalifa University, Doha, Qatar*	82
Mengiste, Shegaw Anagaw	*University College of Southeast, Norway*	29
Moise, Gabriela	*Petroleum-Gas University of Ploiești, Romania*	50
Mușan, Mircea Adrian	*"Lucian Blaga" University of Sibiu, Romania*	60
Săndică, Robert	*"Lucian Blaga" University of Sibiu, Romania*	13
Sângeorzan, Livia	*Transilvania University of Brașov, Romania*	67
Simian, Dana	*"Lucian Blaga" University of Sibiu, Romania*	13
Stoica, Florin	*"Lucian Blaga" University of Sibiu, Romania*	72
Stoica, Laura	*"Lucian Blaga" University of Sibiu, Romania*	72

Streitferdt, Detlef	*Ilmenau University of Technology, Germany*	9
Tuba, Eva	*John Naisbitt University, Belgrad, Serbia*	82
Tuba, Milan	*John Naisbitt University, Belgrad, Serbia*	11
Vlădoiu, Monica	*Petroleum-Gas University of Ploiești, Romania*	50

ISSN 2067-3965