Robertas Damaševičius
Giedrė Vasiljevienė (Eds.)

Communications in Computer and Information Science 920

Information and Software Technologies

24th International Conference, ICIST 2018
Vilnius, Lithuania, October 4–6, 2018
Proceedings

Springer

Communications in Computer and Information Science 920

Commenced Publication in 2007
Founding and Former Series Editors:
Phoebe Chen, Alfredo Cuzzocrea, Xiaoyong Du, Orhun Kara, Ting Liu,
Dominik Ślęzak, and Xiaokang Yang

More information about this series at http://www.springer.com/series/7899

Robertas Damaševičius · Giedrė Vasiljevienė (Eds.)

Information and Software Technologies

24th International Conference, ICIST 2018
Vilnius, Lithuania, October 4–6, 2018
Proceedings

Springer

Editors
Robertas Damaševičius
Kaunas University of Technology
Kaunas
Lithuania

Giedrė Vasiljevienė
Kaunas University of Technology
Kaunas
Lithuania

ISSN 1865-0929 ISSN 1865-0937 (electronic)
Communications in Computer and Information Science
ISBN 978-3-319-99971-5 ISBN 978-3-319-99972-2 (eBook)
https://doi.org/10.1007/978-3-319-99972-2

Library of Congress Control Number: 2018952480

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

Preface

We are happy to present you this book, *Information and Software Technologies*, which is a collection of papers that were presented at the 24th International Conference on Information and Software Technologies, ICIST 2018. The annual conference took place during October 4–6, 2018, in Vilnius, Lithuania.

The book consists of four chapters, which correspond to the four major areas that were covered during the conference, namely, “Information Systems,” “Business Intelligence for Information and Software Systems,” “Software Engineering,” and “Information Technology Applications.” These chapters are further subdivided according to the eight special sessions that were held at the conference. They are the following: (a) Innovative Applications for Knowledge Transfer Support, (b) e-Health Information Systems, (c) Information and Software Technologies for Intelligent Power Systems, (d) Intelligent Methods for Data Analysis and Computer-Aided Software Engineering, (e) Intelligent Systems and Software Engineering Advances, (f) Smart e-Learning Technologies and Applications, (g) Language Technologies, and (h) Digital Transformation.

Every year ICIST attracts researchers from all over the world, and this year was not an exception – we received 124 submissions from more than 30 countries. More importantly, there were participants from many more countries, which indicates that the conference is truly gaining more and more international recognition as it brings together a vast number of brilliant specialists who represent the aforementioned fields and share information about their newest scientific research investigations and the results achieved. Since we always strive to make the conference presentations and proceedings of the highest quality possible, we only accept papers that present the results of various investigations directed to the discovery of new scientific knowledge in the area of information and software technologies. Hence, only 47 papers were accepted for publication (i.e., a 38% acceptance rate). All the papers were reviewed and selected by the Program Committee, which comprised 101 reviewers (together with 74 additional reviewers) from over 90 academic institutions. As usual, each submission was reviewed following the double-blind process by at least two reviewers. When necessary, some of the papers were reviewed by three or four reviewers. Our deepest thanks and appreciation go to all the reviewers for devoting their precious time to produce thorough reviews and feedback to the authors.

We would also like to express our gratitude to the general chair, Prof. Eduardas Bareiša (Kaunas University of Technology), as well as to the session chairs and co-chairs, Prof. Irene Krebs (Brandenburg University of Technology Cottbus-Senftenberg), Prof. Justyna Patalas-Maliszewska (University of Zielona Góra), Prof. Rolf Engelbrecht (ProRec), Assoc. Prof. Vytenis Punys (Kaunas University of Technology), Prof. Giedrius Vanagas (Lithuanian University of Health Sciences), Prof. Algirdas Pakštas (Vilnius University), Assoc. Prof. Vira Shendryk (Sumy State University), Dr. Marcin Woźniak (Silesian University of Technology), Prof. Emiliano

Tramontana and Prof. Christian Napoli (University of Catania), Assoc. Prof. Danguolė Rutkauskienė (Kaunas University of Technology), Prof. Radu Adrian Vasiu (Politehnica University of Timisoara), Prof. Audrius Lopata (Vilnius University), Assoc. Prof. Jurgita Kapočiūtė-Dzikienė (Vytautas Magnus University), Peter Dirix (University of Leuven), Prof. Carsten Wolff and Prof. Christian Reimann (Dortmund University of Applied Sciences and Arts), and Adj. Prof. Pasi Kuvaja (University of Oulu) for their expertise, assistance, and invaluable contribution in making the conference a top-quality scientific event.

In addition, we would like to thank the local Organizing Committee and the Faculty of Informatics, Kaunas University of Technology, for the conference would not have been a great success without their tremendous support. We are also grateful to the Research Council of Lithuania for financial support.

The proceedings of the ICIST 2018 conference are published as a volume of *Communications in Computer and Information Science* series for the seventh time. This would not be possible without the kind assistance provided by Leonie Kunz, Aliaksandr Birukou, and Ingrid Beyer, Springer, for which we are utmost grateful. We are very proud of this collaboration and believe that this fruitful partnership will continue for many more years to come.

July 2018

Giedrė Vasiljevienė
Robertas Damaševičius

Organization

The 24th International Conference on Information and Software Technologies (ICIST 2018) was organized by Kaunas University of Technology and was held in Vilnius, Lithuania (October 4–6, 2018).

General Chair

Eduardas Bareiša	Kaunas University of Technology, Lithuania

Local Organizing Committee

Giedrė Vasiljevienė (Chair)	Kaunas University of Technology, Lithuania
Romas Šleževičius	Kaunas University of Technology, Lithuania
Lina Repšienė	Kaunas University of Technology, Lithuania
Vilma Mikašytė	Kaunas University of Technology, Lithuania
Gintarė Lukoševičiūtė	Kaunas University of Technology, Lithuania
Berta Dauknytė	Kaunas University of Technology, Lithuania

Special Section Chairs

Irene Krebs	Brandenburg University of Technology Cottbus - Senftenberg, Germany
Marcin Woźniak	Silesian University of Technology, Poland
Danguolė Rutkauskienė	Kaunas University of Technology, Lithuania
Audrius Lopata	Vilnius University, Lithuania
Maria Dolores Afonso Suárez	SIANI University Institute, Spain
Rolf Engelbrecht	ProRec Germany, Germany
Justyna Patalas-Maliszewska	University of Zielona Góra, Poland
Emiliano Tramontana	University of Catania, Italy
Christian Napoli	University of Catania, Italy
Carsten Wolff	Dortmund University of Applied Sciences and Arts, Germany
Christian Reimann	Dortmund University of Applied Sciences and Arts, Germany
Michael Reiner	IMC Krems University of Applied Science, Austria
Pasi Kuvaja	University of Oulu, Finland
Jurgita Kapočiūtė-Dzikienė	Vytautas Magnus University, Lithuania
Peter Dirix	University of Leuven, Belgium
Algirdas Pakštas	Vilnius University, Lithuania

Vira Shendryk	Sumy State University, Ukraine
Radu Adrian Vasiu	Politehnica University of Timisoara, Romania
Vytenis Punys	Kaunas University of Technology, Lithuania
Giedrius Vanagas	Lithuanian University of Health Sciences, Lithuania

Program Committee

Irene Krebs	Brandenburg University of Technology Cottbus-Senftenberg
Marcin Woźniak	Silesian University of Technology, Poland
Danguolė Rutkauskienė	Kaunas University of Technology, Lithuania
Rolf Engelbrecht	ProRec Germany, Germany
Emiliano Tramontana	University of Catania, Italy
Radu Adrian Vasiu	Politehnica University of Timisoara, Romania
Vytenis Punys	Kaunas University of Technology, Lithuania
Giedrius Vanagas	Lithuanian University of Health Sciences, Lithuania
Olga Kurasova	Vilnius University, Lithuania
Jurgita Kapočiūtė-Dzikienė	Vytautas Magnus University, Lithuania
Yuh-Min Tseng	National Changhua University of Education, Taiwan
Constantine Filote	Stefan cel Mare University of Suceava, Romania
Marisa Gil	Polytechnic University of Catalonia, Spain
Mehmet Aksit	University of Twente, The Netherlands
Saulius Gudas	Vilnius University, Lithuania
Sanda Martinčić-Ipšić	University of Rijeka, Croatia
Milena Krumova	Technical University of Sofia, Bulgaria
Mirjana Ivanovic	University of Novi Sad, Serbia
Alvydas Jaliniauskas	Harland Clarke Company, USA
Massimo Tivoli	University of L'Aquila, Italy
Raimundas Jasinevičius	Kaunas University of Technology, Lithuania
Damjan Vavpotič	University of Ljubljana, Slovenia
Sandro Leuchter	Hochschule Mannheim University of Applied Sciences, Germany
Martin Gaedke	Technical University of Chemnitz, Germany
John Gammack	College of Technological Innovation, United Arab Emirates
Paulo Rupino da Cunha	University of Coimbra, Portugal
Jyrki Nummenmaa	University of Tampere, Finland
Algirdas Pakštas	Vilnius University, Lithuania
Marcin Paprzycki	Systems Research Institute, Polish Academy of Science, Poland
Janis Grabis	Riga Technical University, Latvia
Ana Paula Neves Ferreira da Silva	University of Coimbra, Portugal
Tor-Morten Grønli	Oslo School of Arts, Communication and Technology, Norway
Christophoros Nikou	University of Ioannina, Greece

Elena Sánchez Nielsen	University of San Fernando de la Laguna, Spain
Vira Shendryk	Sumy State University, Ukraine
André Schekelmann	Niederrhein University of Applied Science, Germany
Virgilijus Sakalauskas	Vilnius University, Lithuania
Dalia Krikščiūnienė	Vilnius University, Lithuania
Audrius Lopata	Vilnius University, Lithuania
Aleksandras Targamadzė	Kaunas University of Technology, Lithuania
Laimutis Telksnys	Vilnius University, Lithuania
Peter Thanisch	University of Tampere, Finland
Lovro Šubelj	University of Ljubljana, Slovenia
Karin Harbusch	University of Koblenz-Landau, Germany
Joao Manuel R. S. Tavares	University of Porto, Portugal
Zakaria Maamar	Zayed University, United Arab Emirates
Juan Manuel Vara Mesa	University of Rey Juan Carlos, Spain
Alexander Maedche	University of Mannheim, Germany
Olegas Vasilecas	Vilnius Gediminas Technical University, Lithuania
Rimantas Butleris	Kaunas University of Technology, Lithuania
Tomas Krilavičius	Vytautas Magnus University, Lithuania
Lucio Tommaso de Paolis	University del Salento, Italy
Eduard Babkin	National Research University, Russia
Tommi Mikkonen	Tampere University of Technology, Finland
Hana Skalska	University of Hradec Králové, Czech Republic
Linas Laibinis	Abo Akademi University, Finland
Benkt Wangler	Stockholm University, Sweden
Kuldar Taveter	Tallinn University of Technology
Kristina Sutiene	Kaunas University of Technology, Lithuania
Janis Stirna	Stockholm University, Sweden
Justyna Patalas-Maliszewska	University of Zelona Gora, Poland
Miloslava Cerna	University of Hradec Králové, Czech Republic
Tomas Blažauskas	Kaunas University of Technology, Lithuania
Prima Gustienė	Karlstad University, Sweden
Wojciech Mitkowski	AGH University of Science and Technology, Poland
Irina Klizienė	Kaunas University of Technology, Lithuania
Renata Burbaitė	Kaunas University of Technology, Lithuania
Jorge Garcia	University of Porto, Portugal
Kęstutis Kapočius	Kaunas University of Technology, Lithuania
Tomas Skersys	Kaunas University of Technology, Lithuania
Martynas Patašius	Kaunas University of Technology, Lithuania
Rytis Maskeliūnas	Kaunas University of Technology, Lithuania
Lina Čeponienė	Kaunas University of Technology, Lithuania
Raimundas Matulevičius	University of Tartu, Estonia
Rimantas Barauskas	Kaunas University of Technology, Lithuania
Rita Butkienė	Kaunas University of Technology, Lithuania
Agnius Liutkevičius	Kaunas University of Technology, Lithuania
Ana Meštrović	University of Rijeka, Croatia

Andrzej Jardzioch	West Pomeranian University of Technology Szczecin, Poland
Tomas Danikauskas	Kaunas University of Technology, Lithuania
Beata Gavurova	Technical University of Košice, Slovakia
Armantas Ostreika	Kaunas University of Technology, Lithuania
Dominykas Barisas	Kaunas University of Technology, Lithuania
Pavel Jirava	University of Pardubice, Czech Republic
Alius Noreika	Kaunas University of Technology, Lithuania
Eva Rakovska	University of Economics Bratislava, Slovakia
Natalia Loukachevitch	Moscow State University, Russia
Gintaras Palubeckis	Kaunas University of Technology, Lithuania
Reima Suomi	University of Liechtenstein, Liechtenstein
Seweryn Spalek	Silesian University of Technology, Poland
Vytautas Štuikys	Kaunas University of Technology, Lithuania
Christian Napoli	University of Catania, Italy
Carsten Wolff	Dortmund University of Applied Sciences and Arts, Germany
Christian Reimann	Dortmund University of Applied Sciences and Arts, Germany
Alfonsas Misevičius	Kaunas University of Technology, Lithuania
Rimvydas Simutis	Kaunas University of Technology, Lithuania
Peter Dirix	University of Leuven, Belgium
Mokhtar Beldjehem	University of Ottawa, Canada
Marjan Mernik	University of Maribor, Slovenia
Liesbeth Augustinus	University of Leuven, Belgium
Nina Tahmasebi	University of Gothenburg, Sweden

Additional Reviewers

Evgeniy Krastev	Sofia University St. Kliment Ohridsky, Bulgaria
Mariya Stankova	South-West University Neofit Rilski, Bulgaria
Jonas Čeponis	Kaunas University of Technology, Lithuania
Vytautas Rudžionis	Kaunas University of Technology, Lithuania
Ludvik Eger	University of West Bohemia, Czech Republic
Alfonsas Vainoras	Lithuanian University of Health Sciences, Lithuania
Wang Gang	Xi'an University of Finance and Economics, China
Tõnu Tamme	University of Tartu, Estonia
Andres Ramirez-Portilla	Universidad Iberoamericana Ciudad de México, Mexico
Pawel Dworak	West Pomeranian University of Technology, Poland
Alessandro Filippeschi	School of Advanced Studies Sant'Anna, Italy
Dariusz Paczko	Opole University of Technology
Kastytis Ratkevičius	Kaunas University of Technology, Lithuania
Slobodan Beliga	University of Rijeka, Croatia
Zenonas Navickas	Kaunas University of Technology, Lithuania
Kristina Bespalova	Kaunas University of Technology, Lithuania

Ilhan Tarimer	Muğla Sıtkı Koçman University, Turkey
Virginija Limanauskienė	Kaunas University of Technology, Lithuania
Daina Gudonienė	Kaunas University of Technology, Lithuania
Vacius Jusas	Kaunas University of Technology, Lithuania
Zbigniew Marszałek	Silesian University of Technology, Poland
Antanas Lenkevičius	Kaunas University of Technology, Lithuania
Marcin Gabryel	Częstochowa University of Technology, Poland
Lina Narbutaitė	Kaunas University of Technology, Lithuania
Solvita Bērziša	Riga Technical University, Latvia
Borislav Djordjevic	Institute Mihailo Pupin, Serbia
Valentina Timcenko	Institute Mihailo Pupin, Serbia
Jordan Hristov	University of Chemical Technology and Metallurgy, Bulgaria
Ranka Stanković	University of Belgrade, Serbia
Volkan Tunali	Celal Bayar University, Turkey
Ramūnas Kubiliūnas	Kaunas University of Technology, Lithuania
Dawid Polap	Silesian University of Technology, Poland
Alberto Rodriguez	Universidad Miguel Hernandez de Elche, Spain
Damian Słota	Silesian University of Technology, Poland
Slawomir Kłos	University of Zielona Góra, Poland
Judita Kasperiūnienė	Vytautas Magnus University, Lithuania
Vytautas Pilkauskas	Kaunas University of Technology, Lithuania
Gytis Vilutis	Kaunas University of Technology, Lithuania
Marta Wlodarczyk-Sielicka	Maritime University of Szczecin, Poland
Sercan Oruc	Middle East Technical University, Turkey
Piotr Artiemjew	University of Warmia and Mazury in Olsztyn, Poland
Evgeny Cherkashin	V.M. Matrosov Institute of System Dynamics and Control Theory, Siberian Branch of Russian Academy of Sciences, Russia
Onder Demir	Marmara University, Turkey
Branislav Popović	University of Novi Sad, Serbia
Mincong Tang	Beijing Jiaotong University, China
Iryna Zolotaryova	Semen Kuznets National University of Kharkiv, Ukraine
Nikolai Scerbakov	Graz University of Technology, Austria
Leon Abdillah	Bina Darma University, Indonesia
Gulistan Raja	UET Taxila, Pakistan
Rasa Bruzgienė	Kaunas University of Technology, Lithuania
Jamshed Iqbal	COMSATS University Islamabad, Pakistan
Necmettin Ozkan	Turkiye Finans Participation Bank, Turkey
Rasa Kasperienė	Vytautas Magnus University, Lithuania
Deniss Stepins	Riga Technical University, Latvia
Vladimir Mityushev	Pedagogical University of Cracow, Poland
Veselina Yankova	Higher Colleges of Technologies, UAE
Algirdas Šukys	Kaunas University of Technology, Lithuania
Ilona Veitaitė	Vilnius University, Lithuania

Radu Ursulean	The Gheorghe Asachi Technical University, Romania
Radek Fujdiak	Brno University of Technology, Czech Republic
Leslaw Golebiowski	Rzeszów University of Technology, Poland
Nuno Castela	Polytechnic Institute of Castelo Branco, Portugal
Josef Basl	University of West Bohemia, Czech Republic
Šarūnas Packevičius	Kaunas University of Technology, Lithuania
George Tambouratzis	Institute for Language and Speech Processing, Athena Research Centre, Greece
Haruko Sanada	Rissho University, Japan
Beata Zielosko	University of Silesia, Poland
Nikša Jakovljević	University of Novi Sad, Serbia
Vladimir Bartošek	Brno University of Technology, Czech Republic
Isaar Arab	Technical University of Munich, Germany
Farid Orujov	Kaunas University of Technology, Lithuania
Andrius Lauraitis	Kaunas University of Technology, Lithuania
Rolandas Girčys	Kaunas University of Technology, Lithuania
Šarūnas Grigaliūnas	Kaunas University of Technology, Lithuania

Co-editors

Robertas Damaševičius	Kaunas University of Technology, Lithuania
Giedrė Vasiljevienė	Kaunas University of Technology, Lithuania

Contents

Business Intelligence for Information and Software Systems: Special Session on Intelligent Methods for Data Analysis and Computer-Aided Software Engineering

Software Engineering: Special Session on Intelligent Systems and Software Engineering Advances

Information Technology Applications: Special Session on Language Technologies

Information Systems: Special Session on Innovative Applications for Knowledge Transfer Support

An Information System Supporting the Eliciting of Expert Knowledge for Successful IT Projects

Justyna Patalas-Maliszewska[1(✉)] and Irene Krebs[2]

[1] Institute of Computer Science and Production Management, University of Zielona Góra, Zielona Góra, Poland
J.Patalas@iizp.uz.zgora.pl
[2] Potsdam University, Potsdam, Germany
irene.krebs@uni-potsdam.de

Abstract. In order to guarantee the success of an IT project, it is necessary for a company to possess expert knowledge. The difficulty arises when experts no longer work for the company and it then becomes necessary to use their knowledge, in order to realise an IT project. In this paper, the ExKnowIT information system which supports the eliciting of expert knowledge for successful IT projects, is presented and consists of the following modules: (1) the identification of experts for successful IT projects, (2) the eliciting of expert knowledge on completed IT projects, (3) the expert knowledge base on completed IT projects, (4) the Group Method for Data Handling (GMDH) algorithm, (5) new knowledge in support of decisions regarding the selection of a manager for a new IT project. The added value of our system is that these three approaches, namely, the elicitation of expert knowledge, the success of an IT project and the discovery of new knowledge, gleaned from the expert knowledge base, otherwise known as the decision model, complement each other.

Keywords: Expert knowledge · IT project · Information system
GMDH

1 Introduction

The process of the elicitation of expert knowledge is a prerequisite for any company, in order to extract the best of workers' know-how, so as to be able to store it, in the case of employees' holiday leave or other absences from work. The process of the elicitation of expert knowledge is carried out through the medium of questionnaires or interviews where experts provide qualitative answers to an analyst [1]. Based on these qualitative answers, results can therefore be both a mixture of the views of the experts and the views of the analysts [2]. We can also distinguish a method for the elicitation of expert knowledge from such as cognitive maps [3], diagrams [4] and up to and including "brainstorming" [5] as well as the traditional Delphi Process.

The main player in the process of the elicitation of expert knowledge is the actual expert himself/herself, so all eyes should be on selecting experts with the "best" knowledge [6].

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 3–13, 2018.
https://doi.org/10.1007/978-3-319-99972-2_1

Company workers involved in software projects, require ready access to knowledge [7] and the key sources for this knowledge are the experts [8]. Our research question is firstly, therefore, how to select an expert from within a company; secondly, how then to elicit his/her knowledge and finally how, then, to utilise the new and knowledge usefully, in the context of successful IT projects.

In this paper, the concept and implementation of an information system, supported by the elicitation of expert knowledge (ExKnowIT) for successful IT projects, is proposed. We know that the success of an IT project can be evaluated using different methods and tools [9] therefore we can also develop, as part of our information system, a decision model supporting the selection of a manager for a new IT project based on the GMDH method, which was created and based on expert knowledge. Our system is also investigated in the form of a web-application and is presented, based on a case study.

Section 2 shows the research literature related to the approach for the elicitation of expert knowledge and the determining of successful IT projects and describes the modules which are involved in ExKnowIT. Section 3 presents the exemplar functionalities of ExKnowIT, especially the decision support model, based on the GMDH method. Section 4 summarises the research results.

2 Expert Knowledge for Successful IT Projects and an Approach to ExKnowIT

According to the research literature of Sanchez and Terlizzi [10] we know that project size, project duration, postponement, team size, team allocation and team diversity, all determine successful IT projects. Gingnell et al. (2014) distinguish the most important factors for the success of IT projects as being user involvement, the support of executive management, clear goals and objectives, project management skills and internal communications [11]. We know, also, that the success of a project focusses on its budget, its schedule and its specifications [12].

The success of an IT project depends, to a large extent, on expert knowledge which is a pre-requisite for any company with difficulties which have arisen when experts no longer work for a company but where their knowledge is still necessary for the completion of an IT project.

The success of IT projects needs to be supported, therefore, from two dimensions; one which focusses on expert knowledge and the other which exhibits all the characteristics of those IT projects, previously completed. Eliciting of Expert Knowledge process as the approach to our system is presented in the Fig. 1.

Our information system: ExKnowIT, supported by the elicitation of expert knowledge (Fig. 1), therefore, consists of the following elements:

(1) *Identification of experts for successful IT projects*

The process for the elicitation of experts within an IT company is carried out via web-questionnaires. Each employee completes web-based questionnaires, according to the authors' algorithm, with special emphasis on the function of personal usefulness and its components, in such as general knowledge, professional knowledge,

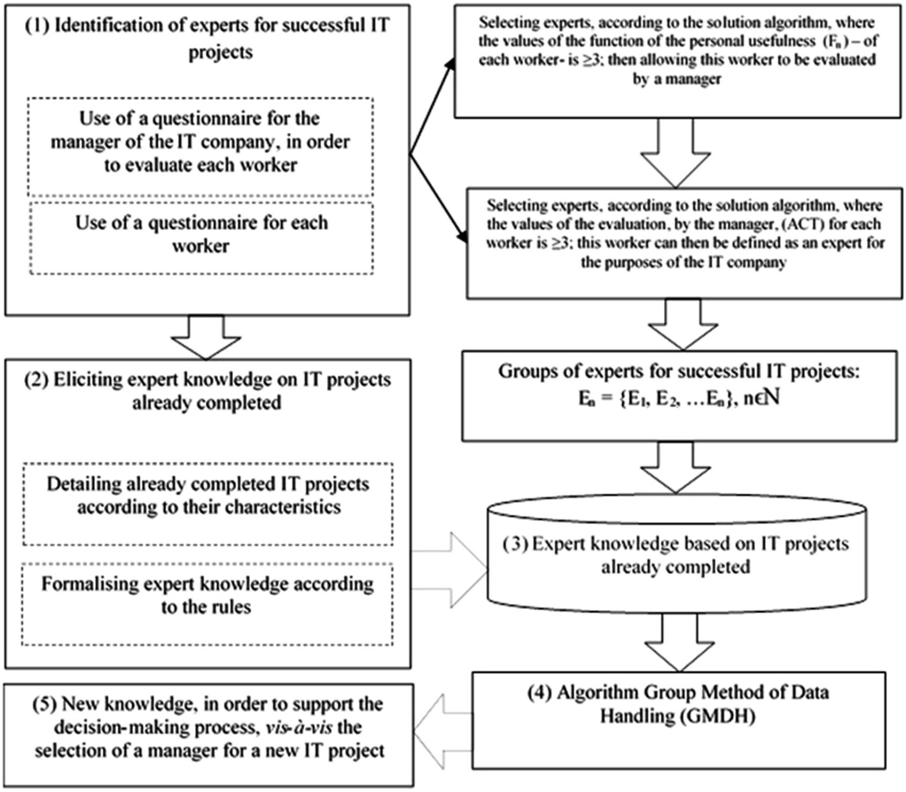


Fig. 1. Eliciting of expert knowledge

professional abilities, experience and the capacity for innovation [13–15] the value of this function may then be determined for each employee.

Next, if the values of the function of personal usefulness (F_n) of each worker is ≥ 3, the manager should then be able to evaluate this worker. Furthermore, if the values of the evaluation, by a manager (ACT) for each worker is ≥ 3 we can then define this worker as an expert, for the purposes of the IT company.

(2) *Eliciting expert knowledge on completed IT projects*

For each IT project completed within a company, the manner in which it is reported is created according to the following characteristics of the IT project itself, namely, identification of the customer regarding the industry in question, its size, its requirements and any training details; identification of the project, namely its goal, location, scope, work/team work, methodology, communication rules, duration, the programming-language used and reports of such as mistakes. Reports on IT projects completed are created by an identified expert from the IT company.

(3) *The expert knowledge base*

The expert knowledge base is compiled, by experts from the answers received from the questionnaires sent out regarding completed IT projects. Each filed questionnaire is then converted to the knowledge base according to the following rules (Table 1):

Table 1. Rules for creating the expert knowledge base

Questionnaire replies/values in the expert knowledge base	1	2	3
Customer identification: industry (CI)	A manufacturing company	A service company	A manufacturing company, as well as a service company
Customer identification: size (CS)	A small company	A medium company	A large company
Customer identification: requirements (CR)	The implementation of an application	The creation of a new solution for a customer	Improvement in the existing functionality of solutions, in an application, or if for the addition of current, functionality solutions to an application
Customer identification: training (CT)	The users of the application had been trained in the Head Office	The users of the application had been trained at the Client's Office	The users of the application had been trained *via* a network, such as the Internet
Project identification: goal (PG)	The implementation of an application	The creation of a new solution for a customer	The improvement of the functionality of existing solutions
Project identification: scope (PS)	The implementation of a CRM and/or for Sale and/or for Personnel and/or for Finance	The implementation of Production and/or for Logistics and/or for Supplies and/or for Warehouse	The implementation of e-commerce and/or Business Intelligence and/or B2B and/or B2C
Project identification: location (PL)	At Head Office	At the Client's Office	At both the Head Office and also at the Client's Office
Project identification: work (PW)	Employees within the company only	Consultants from inside the company's environment only	Employees, who are engaged in working at the customer's only
Project identification: team work (PT)	Individual work	Team work	Both individual work and team work

(*continued*)

Table 1. (*continued*)

Questionnaire replies/values in the expert knowledge base	1	2	3
Project identification: methodology (PM)	The methodology is classic	The methodology is agile	The methodology is both classic and agile
Project identification: duration (PD)	Less than 3 months	3–12 months	Over 12 months
Project identification: communication (PC)	*via* email and/or Chat and/or video-conferencing	*via* direct conversation and/or meetings	*Both via* email and/or Chat and/or video-conferencing and *via* direct conversation and/or meetings
Project identification: programming – language (PP)	Programming – Java Script	Programming – PHP	Another: (e.g. Language: Python or Language: C (C++) or (C#) or Objective-C or programming – Language: (Visual) Basic or programming – Language: Perl or programming – Language: Delphi/Object Pascal or programming – Language: Visual Basic.NET or programming – Language: Assembler or programming – Language: PL/SQL or programming – Language: Swift or programming – Language: MATLAB or programming – Language: Groovy
Project identification: reports (PR)	Mistakes in customer requirements	Errors in the application	The report is about mistakes in the project or about changes to the project

(4) *GMDH algorithm*

The GMDH algorithm can be treated as a genetic algorithm with the following interpretation: a chromosome is a single, polynomial, the population is a set of polynomials, considered in the current iteration, the function of evaluation (fit) is a regularity criterion. Accepting the output data as the input data of the next generation of polynomials and creating a new polynomial from two, partial polynomials, can be interpreted as crossing, in the sense of genetic algorithms. A certain modification in relation to the typical, genetic algorithm is that all the population chromosomes are exchanged in one iteration. The GMDH algorithm offers a wide range of possible applications due to the considerable freedom in defining certain elements of the synthesis. The use of classic solutions for the algorithm discussed, gives the possibility of estimating static systems and automatic control systems. Thanks to the introduction of certain generalisations in the definition of the input values and the procedure for their selection, the GMDH algorithm can also be used in relation to understanding knowledge and - in our case- to the determining of new knowledge. Determining the knowledge structure by means of the GMDH algorithm relies on the iterative repetition of a specific sequence of operations, leading to an evolutionary resultant structure. This process is terminated when the optimal degree of complexity is reached [16, 17]. The result of the algorithm's operation is a polynomial which is the model of the object.

(5) *New knowledge in support of a decision about the selection of a manager for a new IT project*

With regard to arriving at a decision model about the selection of a manager for a new IT project, the characteristics of IT projects (IT_o) and of the evaluation of successful IT projects by experts, are described:

- the evaluation of successful IT projects by experts, where, firstly, the IT project is a complete success, in accordance with the planned time, budget and the scope; where, secondly, there is partial success of the IT project, there being a degree of inconsistency with one of the parameters of the IT project, in such as the planned time, the budget and the scope and thirdly, where the IT project is unsuccessful, the planned time, the budget and the scope.
- Characteristics of IT projects: IT_o = {CI, CS, CR, CT, PG, PS, PL, PW, PT, PM, PD, PC, PP, PR}, where CI - customer identification: industry, CS - customer identification: size, CR - customer identification: requirements, CT - customer identification: training, PG - project identification: goal, PS - project identification: scope, PL - project identification: location, PW- project identification: work, PT - project identification: team work, PM - project identification: methodology, PD - project identification: duration, PC - project identification: communication rules, PP -project identification: programming – language, PR - project identification: reports.

The limitation is that an IT project, with designated characteristics, should be provided by an expert, in order to guarantee the success of this project. Therefore, an identified expert in a company, is constantly evaluating each IT project.

The decision model is thus created, using the GMDH algorithm, based on expert knowledge:

$$IT_s = A + B \times IT_{op} + C \times IT_{oq} + D \times IT_{op}^2 + F \times IT_{oq}^2 + G \times IT_{op} IT_{oq} \tag{1}$$

where: IT_s – new knowledge about the success of an IT project, IT_{op}, IT_{oq} - indicators which describe the characteristics of the IT projects IT_{op}, IT_{oq} $\in$ $\leq$ 0;3 $\geq$, A,B,C,D, F,G – estimators.

In order to facilitate the description of the defined model, evaluation of the IT projects should be standardised according to the values of the experts as shown:

$$E = \begin{bmatrix} E_1 \\ E_2 \\ \ldots \\ E_n \end{bmatrix} \tag{2}$$

where n $\in$ N.

In this definition, E_i is the i-th E. Also included in the matrix are the indicators which describe the evaluation of the IT projects. Each E_i is associated with indicators that are characteristics of IT projects. A characteristic of the output of IT projects is provided, in order to show the relationship between the success of the IT project and the outcome of completed IT projects. Therefore, the characteristics of the output of IT projects provides a set of indicators, noted as:

$$IT_O = \begin{bmatrix} CI_1 & CS_1 & CR_1 & CT_1 & PG_1 & PS_1 & PL_1 & PW_1 & PT_1 & PM_1 & PD_1 & PC_1 & PP_1 & PR_1 \\ CI_2 & CS_2 & CR_2 & CT_2 & PG_2 & PS_2 & PL_2 & PW_2 & PT_2 & PM_2 & PD_2 & PC_2 & PP_2 & PR_2 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ CI_n & CS_n & CR_n & CT_n & PG_n & PS_n & PL_n & PW_n & CS_n & PM_n & PD_n & PC_n & PP_n & PR_n \end{bmatrix} \tag{3}$$

where n$\in$N and we state, that each n-expert reports on each n-realised IT project. Each characteristic of the IT project is formalised, according to the rules for - and in the context of - the answers to each question in the web-questionnaire, for reporting the completed IT project.

The decision model on the selection of a manager, based on previously realised IT projects for new IT, was built using the Group Method of Data Handling. Multi-level GMDH allows the optimised synthesis of a mathematical model, for a given class of regression functions which can be used in both evaluating criteria and in quality assessment. Both elements of the algorithm (the values for the success of an IT project and the values of the characteristics of the IT projects) are defined arbitrarily by the author.

3 An Information System Supporting the Eliciting of Expert Knowledge for Successful IT Projects

Below is an extract from ExknowIT, based on the proposed approach and on a real case study from an IT company (see Fig. 1).

According to the first element in ExknowIT, web-knowledge questionnaires are defined as the identification of experts in successful IT projects; this applies to both the workers and to the managers. Figure 2 presents an example of an extract from the web-questionnaire for workers, in order to determine the personal usefulness value of each worker.

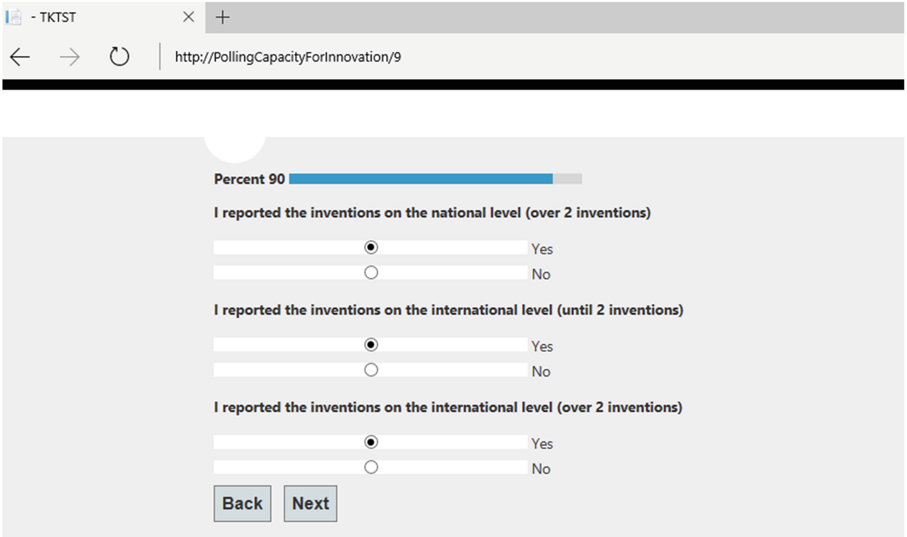


Fig. 2. An extract from the web questionnaire for employees – the Capacity for Innovation component

The personal usefulness value of each worker is obtained based on each completed web-knowledge questionnaire. Thus, if the value of the function of the personal usefulness (F_n) of each worker is ≥ 3, then this worker is to be evaluated by a manager (see Fig. 3 – an example).

Each identified expert is capable of completing reports on completed IT projects: (see Fig. 4).

Based on reports received from experts, expert knowledge is formalised according to defined rules. In accordance with the data received from 5 experts on 5 successfully completed IT projects, all the variations of the GMDH algorithms were subsequently investigated. As a result, the best possible polynomial was obtained, the algorithm evolution process having been completed on the second iteration – see Fig. 5.

$$IT_s = -2.68 - 5.56 \times CI + 8.33 \times PR + 1.18 \times CI^2 - 0.36 \times PR^2 - 0.36 \times CI \times PR \tag{4}$$

where,

- IT_s - the statement of success of the completed IT project,

- CI - the characteristics of the IT project: customer identification: industry, CI $\epsilon \leq 0;3 \geq$.
- PR- the characteristics of the IT project: project identification: reports, PR $\epsilon \leq 0;3 \geq$.

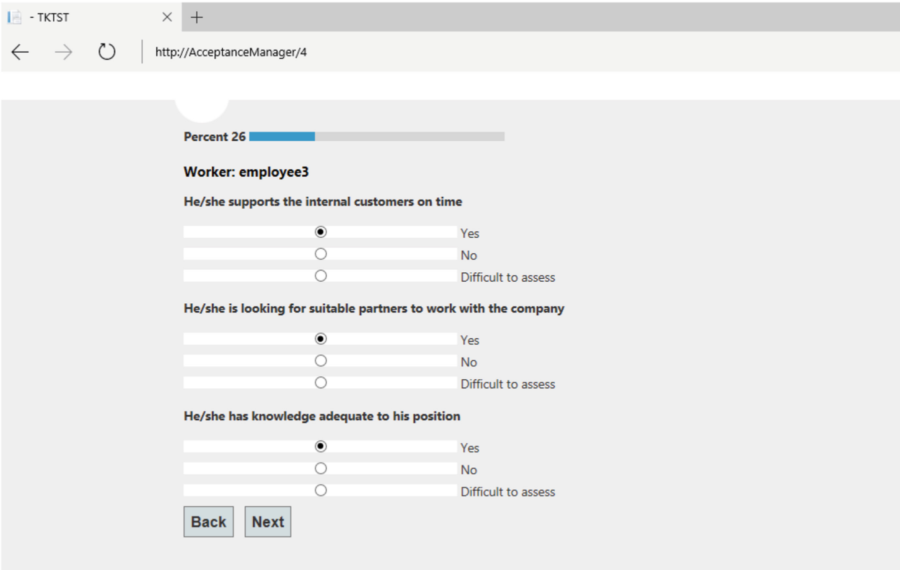


Fig. 3. Extract from the managers' web-questionnaire, for the evaluation of an employee

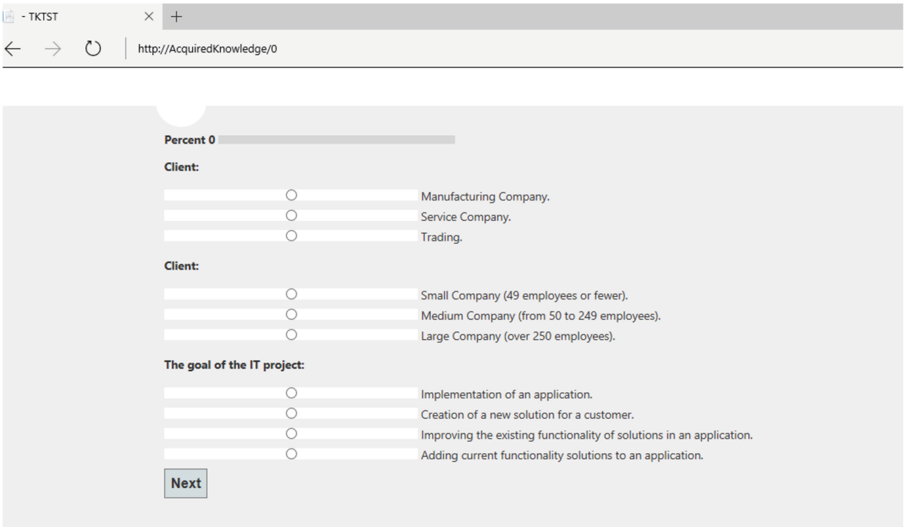


Fig. 4. An extract from a report on an IT project

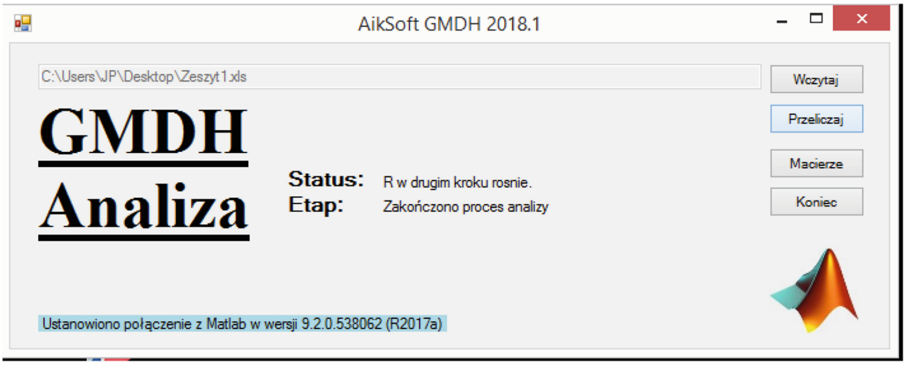


Fig. 5. Variations of the algorithm, Group Method of Data Handling (GMDH) for the expert knowledge base in an IT company

Based on the ITs model obtained (formula 4), it is therefore possible to simulate the possibility of achieving success in completing an IT project in a manufacturing company. Using this model, we can state that the most significant impact on the success of a new IT project, is due to the fact that ongoing reports were prepared in the customer's industry, during completion of IT projects. We also know that the highest value of the ITs model is for CI = 1 i PR = 3. Therefore, in order to achieve success in an IT project, the customer should be a manufacturing company and the expert, managing this project, must include in his/her reports, all the changes made during completion of the IT project; furthermore, only those managers should be selected, who report all changes during the provision of new IT projects for the manufacturing company.

4 Conclusions

Our ExKnowIT information system, supporting the eliciting of expert knowledge for successful IT projects, consists of the following modules: (1) identification of experts for successful IT projects, (2) the eliciting of expert knowledge about completed IT projects, (3) the expert knowledge base on completed IT projects, (4) the Group Method of Data Handling (GMDH) algorithm and (5) new knowledge, supporting decisions on the selection of a manager for new IT projects. It was presented, based on a case study which provided an opportunity to elicit expert knowledge for the improvement of successful IT projects.

ExKnowIT holds knowledge elicited from experts. This expert knowledge base is represented by rules, with this formalised expert knowledge and GMDH method then being used to build a decision model about the selection of a manager for a new IT project.

The added value of our system is that these three approaches, viz., the eliciting of expert knowledge, the success of an IT project and the discovery of new knowledge, that is, the decision model from the expert knowledge base, all complement each other. According to Wang et al. [18] the approach can be used to build better knowledge-based systems.

References

1. O'Hagan, A., Buck, C.A., Daneshkhah, A., Eiser, J.R., Garthwaite, P.H., Jenkinson, D.J., Oakley, J.E., Rakow, T.: Uncertain Judgments: Eliciting Experts' Probabilities. Wiley, Chichester (2006)
2. Mäntyniemi, S., Haapasaari, P., Kuikka, S., Parmanne, R., Lehtiniemi, M., Kaitaranta, J.: Incorporating stakeholders' knowledge to stock assessment: central Baltic herring. Canadian J. Fish. Aquat. Sci. **70**, 591–599 (2013)
3. Eden, C.: Cognitive mapping. Eur. J. Oper. Res. **36**(1), 1–13 (1988)
4. Howard, R.A., Matheson, J.E.: Influence diagrams. Decis. Anal. **3**, 127–143 (2005)
5. Rowe, G., Bolger, F.: Final report on the identification of food safety priorities using the Delphi technique. EFSA Supporting Publ. **13**(3), 139 (2016)
6. Bolger, F., Wentholt, M.: Principles and practice of selecting and motivating experts: guidance on expert knowledge elicitation in food and feed safety risk assessment. Eur. Food Saf. Authority (EFSA) J. **12**(6), 138–162 (2014)
7. Rajlich, V.T., Bennett, K.H.: A staged model for the software life cycle. Computer **33**(7), 66–71 (2000)
8. Licorish, S.A., MacDonell, S.G.: Understanding the attitudes, knowledge sharing behaviors and task performance of core developers: a longitudinal study. Inf. Softw. Technol. **56**(12), 1578–1596 (2014)
9. Thomas, G., Fernández, W.: Success in IT projects: a matter of definition? Int. J. Proj. Manag. **26**(7), 733–742 (2008)
10. Sanchez, O.P., Terlizzi, M.A.: Cost and time project management success factors for information systems development projects. Int. J. Proj. Manage. **35**(8), 1608–1626 (2017)
11. Gingnell, L., Franke, U., Lagerström, R., Ericsson, E., Lilliesköld, J.: Quantifying success factors for IT projects-an expert-based Bayesian model. Inf. Syst. Manage. **31**(1), 21–36 (2014)
12. Basten, D., Joosten, D., Mellis, W.: Managers' perceptions of information project success. J. Comput. Inf. Syst. **52**(2), 12–21 (2011)
13. Patalas-Maliszewska, J.: Knowledge Worker Management: Value Assessment, Methods, and Application Tools. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36600-0
14. Patalas-Maliszewska, J., Krebs, I.: Modelling an Application for Tacit Knowledge Acquisition Support for an IT Company. TRENDS UND COMMUNITIES DER RECHTSINFORMATIK. IRIS 2017, Tagungsband des 20. Internationalen Rechtsinformatik Symposions, pp. 275–282. Salzburg, Austria (2017)
15. Patalas-Maliszewska, J., Krebs, I.: A model of the tacit knowledge transfer support tool: CKnow-board. In: Dregvaite, G., Damasevisius, R. (eds.) Information and Software Technologies: 22nd International Conference, ICIST 2016, vol. 639, pp. 30–41. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-319-46254-7_3
16. Farlow, S.J.: Self-organizing Methods in Modelling: GMDH-type Algorithms. Marcel Dekker Inc, New York (1984)
17. Mrówczyńska, M.: Elements of an algorithm for optimizing a parameter-structural neural network. Reports on Geodesy and Geoinform. **101**(1), 27–35 (2016)
18. Wang, X., Qu, H., Liu, P., Cheng, Y.: A self-learning expert system for diagnosis in traditional Chinese medicine. Expert Syst. Appl. **26**, 557–566 (2004)

Metis: A Scalable Natural-Language-Based Intelligent Personal Assistant for Maritime Services

Nikolaos Gkanatsios(✉), Konstantina Mermikli, and Serafeim Katsikas

METIS Cyberspace Technology SA,
4 Sachtouri & 2 Posidonos Ave., 17674 Kallithea, Greece
nickganats@yahoo.gr, {konstantina.mermikli,serafeim.katsikas}@metis.tech

Abstract. We implement an intelligent personal conversational assistant, communicating in natural language and designed specifically for the maritime industry. A multi-stage message analysis is performed, first classifying the topic of the request and finally applying special parsers to extract the parameters needed to execute the task. Our system is scalable and robust, employing generic and efficient algorithms. Our contributions are manifold. First, we present a complex and multi-level natural-language-processing-based system, focused particularly on the maritime domain and incorporating expert knowledge of the field. Next, we introduce a series of algorithms that can extract deep information using the syntactic structure of the message. Lastly, we implement and evaluate our approach, testing and proving our system's effectiveness and efficiency.

1 Introduction

In an era characterized by rapid technological evolution and massive data acquisition, humans find it progressively harder to compete with the increasing complexity of handling immense volumes of information. In maritime industry especially, employees face huge amounts of data that need to be processed in order to extract meaningful inference for vessels' efficiency every day, striving to compete with these challenges. To this end, recent years have seen the development and release of numerous conversational and virtual assistants [1,2] designed for a wide range of fields, that apply high computational power and smart, effective algorithms to organize, process and visualize useful, yet hard to cope with, data. Nevertheless, despite speech being the most prominent communication medium for humans, efficient human-computer interaction through natural language remains very challenging due to its wide diversity. Machine Learning offers an appealing approach, building models that learn automatically from training data, concluding to more robust predictions.

Motivated by the challenges of implementing a robust and useful conversational agent and especially by the lack of other similar work specializing in the maritime field, we apply these Machine Learning approaches in maritime communications. Our chatbot, Metis, is specifically designed for Maritime and is

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 14–28, 2018.
https://doi.org/10.1007/978-3-319-99972-2_2

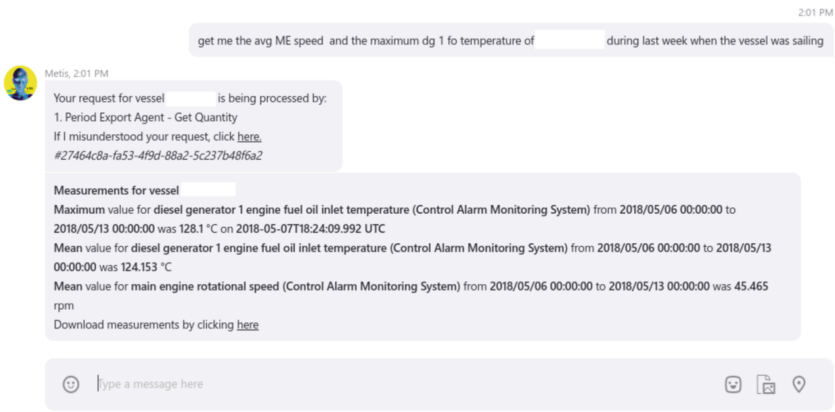


Fig. 1. A real running example of a question-answer conversation with Metis. A classifier identifies the topic and forwards the message to the correspondent expert agent (here "Period Export Agent"). The agent calls special parsers to extract other available information, executes the task and finally responds to the user. Note that the vessel's name is not shown for privacy reasons.

trained with actual data to recognize messages directly in natural language and analyze complex requests in multiple levels, making use of a generic classifier and special parsers. The classifier is employed to predict a message's topic, choosing from a predefined set that covers the core interests of maritime procedures and then forward the request to expert software agents. These programs, employ special parsers to extract all the necessary information, in order to execute the task, create an answer and send it to the users' E-mail, Skype or any other communication channel of their choice (Fig. 1). New functionalities can be added with minimum effort, as our classifier models are fully retrainable and parsers can effectively handle large message variations, therefore providing scalability to our system. The extended use and feedback we receive prove the quality and the importance of our system's results over traditional methods that have been applied in Maritime during the past years.

The rest of this paper is organized as follows: Sect. 2 presents related work on the field of chatbots and message handling using Natural Language Processing (NLP) techniques. Section 3 presents the full system design, while Sect. 4 analyzes the algorithms we implemented. In Sect. 5 we evaluate our system and its components and Sect. 6 summarizes our work and proposes possible future directions.

2 Related Work

Chatbots are recently an emerging technological field with plenty of applications [3–5]. NLP is the most commonly used tool in modern conversational systems' design [6], combined with rich linguistic rules [7] or Machine Learning approaches [8]. Most of these approaches are task-oriented, while others [9] emphasize on maintaining a natural dialog or creating a bot persona [10]. Despite the domain-specific of our approach, we are mostly inspired by [11], which introduces a generic, open-domain NLP-based design for chatbots that performs a three-stage analysis: utterance understanding, dialog control and utterance generation. We differ from this model in terms that we consider that our chatbot can only handle questions and answer back; so the dialog maintains a strict flow.

Utterance understanding often involves a classification into a set of known categories, a well-studied problem [12]. Closest to our approach is the work of [13] that performs a preprocessing and vectorization of the text and then trains a classifier. Message preprocessing includes spelling correction [14,15] and message simplification [16]. We focus on splitting the message upon conjunctions, with the most similar works to ours being those using Dependency Types [17] and Trees [18,19].

A chatbot's answer summarizes the execution of the task it just performed, whether this concerns greeting the user or responding to a complex query. Such an execution on a task-oriented system should be able to decide whether a message provides or demands information. Some approaches focus on question detection [20,21], while others [22] view the problem as distinguishing between imperative and declarative sentences. Closest to ours is the approach of [23] that focuses on questions on Twitter. Lastly, information extraction is an imperative step in order to specify a query. Some interesting works include template matching [24] and learning from stated relationships between specific parsing entities [25]. The chatbot finally uses the extracted information to construct a response.

3 Scalable Multi-level Message Analysis

3.1 Overview

Metis accepts messages from different communication channels (Skype, E-mail, Web Chat and more), using [26], directly in natural language (English) and processes them in multiple stages. First, the message is parsed and preprocessed before it is classified to one or more of the categories that Metis can handle. Currently, there are 35 categories that include requests for information about a vessel's operational status, fuel oil consumption and details about vessel's main engine, but also getting measurements or setting values for specific vessel quantities. Messages for an informal dialog with the bot are also handled. The message is then forwarded to an expert intelligent agent, a programming class that is the specialist responsible for executing the task and creating an answer for the user. The agent employs special parsers to extract the information needed from the user's message, like the vessel quantities and the filters to be applied on measurements. In case important information is missing, our system tries to reconstruct it, otherwise it prompts the user with a message to provide the necessary information. A block diagram of our full system is presented in Fig. 2.

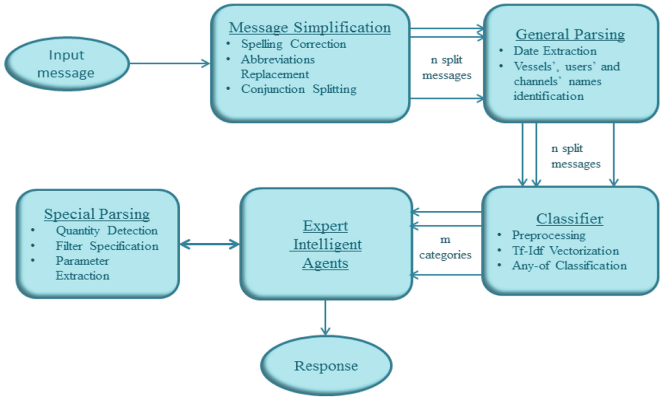


Fig. 2. The implemented system architecture for multi-level message analysis and information extraction.

3.2 Message Simplification

A message in natural language is not always easy to understand, unless some processing and simplification occurs. First of all, mistyping and misspelling errors are common in dialog systems [27,28], especially when most users are maritime employees from different parts of the world, with often poor English skills [3,29], being a serious cause of accidents in shipping processes [30]. Furthermore, maritime language makes common use of abbreviations and shortenings of words and phrases. Lastly, complex sentence structures, that appear when multiple requests are combined, can be both hard to classify and to parse. To address the above challenges, the input message passes through a custom spelling corrector that also embeds a special maritime abbreviations handler (see Sect. 4.3). The postprocessed message is parsed and split upon conjunctions of main verbs and main nouns, using the algorithm we present in Sect. 4.4. Relative and adverbial clauses are transferred between neighboring subsentences that do not contain a clause, in order to avoid any loss of useful information.

3.3 General Parsing

The general parser analyzes the set of messages produced after conjunctions are split, to extract information that is most probably useful for the majority of Metis's agents. This includes the vessel this message refers to, the user that should expect the answer, the communication channel this answer shall be sent and the date or period the request is about. Parameters are shared between messages that were generated by the same conjunction to ensure information is not lost. Moreover, the appearances of these parameters are replaced by codewords ("vessel-name", "username", "channel" and "date" respectively) that filter the information the classifier has to examine, while preserving the message's syntax. Names (vessels, users and channels) are simply detected with a string search, while dates are extracted using a much more complex algorithm we present in detail in Sect. 4.2. By default, the parser expects a message is referring to current time and the user that sent it expects an answer in the channel they used, unless otherwise specified. Lastly, vessel names are temporarily saved in an external database so that the system can retrieve them to continue a conversation. It is left to each agent to decide if more information is demanded and prompt the user with an appropriate message.

3.4 Classifier

In order to find the most appropriate agent to forward the message, a set of classifiers is employed. First, we identify if a message provides or asks for information, using the approach presented in Sect. 4.8. We further distinguish messages requesting information, with applying pure content multi-label classifiers. For this purpose, the message is preprocessed to simplify its content and filter unnecessary or distractive information, before it is transformed into a real-valued sparse vector using Tf-Idf vectorization schema [31]. Preprocessing includes tokenization, removal of unnecessary words (stop words, numbers and codewords) and lemmatization. Finally, adverbial and relative clauses are removed as the topic has to be concluded by the main sentence.

We use the real-valued vectors to train and combine three classifiers, two strong and one weak [32], using Scikit-Learn library [33]. Strong classifiers (Support Vector Machine and Random Forest classifier) are trained to generalize well with vectors that are computed after the stop words' removal, while the weak one (Random Forest again) does not consider stop words' removal and is able to overfit the training data. During classification, strong classifiers vote for cases when the input vector has more than one non-zero entry after stop words are removed, else, only the weak classifier is applied. Final labels are obtained applying the OR function between the predictions of the three classifiers. The scoring metric to be maximized during training is the precision, to reduce false positives when the three classifiers vote and to benefit from the complementary nature of the different algorithms.

3.5 Expert Agents and Special Parsing

A classified message is forwarded to the intelligent agents that are responsible for its execution and the response to the user. These programs employ databases and efficient scientific algorithms to embody expert knowledge into a solution. The parameters for task execution and computations are often specifically requested in the user's message and the agents employ special parsers to extract this information. Common cases are the detection of a name from a given list and the analysis of a conditional or temporal filter of measurements (e.g. *"Plot speed during last week, when wind speed was above 5 beauforts"*), using algorithms described in Sect. 4. The agents then are able to execute the task requested and construct a response. Since this paper focuses on the natural language interface of Metis, the exact agents' architecture and response construction are not presented in more detail.

4 Multi-level Message Analysis Algorithms

4.1 Subtree Generation

SpaCy parser applies Deep Neural Networks to produce dependency trees that represent the relations between words of parsed messages, as in Fig. 3. Each tree has a root, that is usually the main verb or the main noun, if any. In many cases, it is useful to isolate a certain part of a sentence, e.g. an adverbial clause. We view this problem as a subtree generation, where, starting with a given chunk, we recursively append the children of all leaves of the subtree that are not already included in it and that obey to a certain condition (e.g. not being main verbs), until there are no more children to add. The algorithm can be expressed as in Algorithm 1 and is heavily used by plenty of our syntax-based algorithms.

Algorithm 1. Subtree Generation Algorithm

```
function SubtreeGeneration(Tree, Roots, Condition)
    newSubtree ← Roots
    oldSubtree ← [ ]
    repeat
        oldSubtree ← newSubtree
        for node ∈ Tree do
            if node ∉ newSutree AND parent ∈ newSubtree AND Condition(node) then
                Append node to newSubtree
    until newSubtree == oldSubtree
    return newSubtree
```

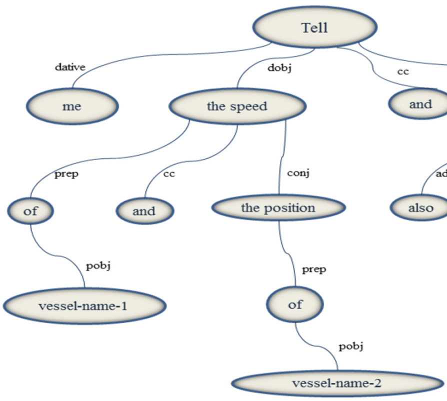


Fig. 3. The dependency tree [34] for the complex sentence "Tell me the speed of vessel-name 1 and the position of vessel-name 2 and send them to my email", as obtained by SpaCy parser [35].

4.2 Date Extraction

Dates are a highly important parameter, yet difficult to extract, as they do not follow any syntactic pattern and they appear with a lot of variations, cross-references (e.g. *"from the 3rd to the 5th of June"*), and ambiguities, especially in maritime communications where dates are seldom written explicitly (e.g. *"11122017"* is a common format for the 11th of December of 2017). We propose an approach, robust against all the above challenges, that uses Natty library [36] as a simple date detector and we extend time point detection to time period identification as well, searching for interval formats (e.g. *"from DATE to DATE"*). To improve Natty's accuracy, the message is processed to transform uncommon date formats into a recognizable form. Ambiguities are resolved by incorporating the sentence's main verb's tense and selecting the date that is closer to current time. Finally, cross-references are handled comparing the information provided in each one of the relative dates and adding the missing values to the less explicit date between the two.

4.3 Spelling Correction and Abbreviation Replacement

Word shortenings and extensive use of abbreviations are more than common in maritime language, while the misspelling rate is also not to be neglected (see Sect. 3.2). We built a corrector that integrates the open source spelling checker PyEnchant, a maritime abbreviations handler and a named entity recognizer, while also being able to handle concatenated words. To this end, we recursively check for possible splits of a word into multiple words that are either valid spellings, abbreviations or names. We extend the list of names by incorporating a special word acceptor for words that are at most one edit away from a word that is part of a vessel's or user's name. If no split can be performed, we keep the most probable correction proposal by PyEnchant.

Abbreviations are then replaced with the help of an extensible list of abbreviations collected by specialists of maritime communications. As the same abbreviation can have different meaning (like "MT" that unfolds either to "metric tonnes" or to "moving tanker"), we also consider neighboring words in order to resolve ambiguities.

4.4 Conjunction Splitting

Splitting a sentence with a complex syntactic structure upon conjunctions to produce a set of sentences with simpler syntax, while fully maintaining the meaning of the original message, has a major role in message comprehension. We view this problem as a generation of a set of independent trees, from an initial complex dependency tree, where terms related with a conjunction are considered equivalent. These terms (in case of main verbs and main nouns) may also share parameters (e.g. clauses, direct object relations for verbs) unless they are explicitly declared for each term separately.

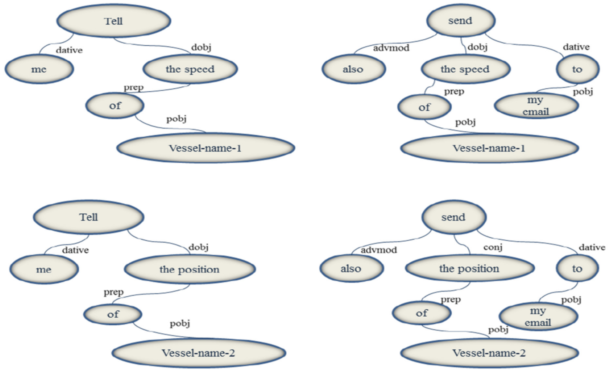


Fig. 4. Illustration of conjunction splitting on the tree of Fig. 3. We obtain four independent sentences, where the pronoun *"them"* has been replaced by the explicit objects it is referring to.

The first step of our algorithm is to split main verbs, which are identified as the root of the sentence, if it is a verb, together with any verbs that are connected to the root with a conjunction. Using main verbs as roots, we create independent subtrees using Algorithm 1 with the condition that main verbs cannot be added to the subtree. Direct objects in third-person-pronoun-form are replaced with the direct object they refer to. Similarly, we split main nouns, which can be identified as the root of the sentence, if it is a noun, or direct objects of the main verb, together with any nouns that are connected to a main noun with a conjunction. A use-case is shown in Fig. 4.

On its last step, the proposed algorithm handles conjunctions between other words that share the same part-of-speech tag. In this case, the conjunction terms are considered fully equivalent and they are forced to inherit each other's children.

4.5 Quantity Detection

A common case in information extraction from text, is searching for a list of names we will refer to as "quantities". Here, multiple challenges arise, as synonyms, shorter names than the one expected, permutations and interferences with irrelevant words may lead to parsing failures. Our quantity detection algorithm faces these challenges by viewing each quantity as a set Q of words q_i with no particular order and different importance w_i to the detection score. Stop words have zero importance, while nouns and adjectives have variable importance, determined by experts of maritime language.

Given a quantity Q of N words and a message of L words, our algorithm computes the similarity scores s_{ij} between q_i and j-th word of the input message, for every $i = 1, \ldots, N$ and $j = 1, \ldots, L$. Next, we "slide" the quantity upon the message with stride 1, examining successive overlapping neighborhoods of $N + margin$ words that are not stop words. The parameter $margin$ is used to extend the number of words in an examining neighborhood in order to handle cases when extra words interfere the quantity words. The local detection score [24,37] of a quantity word in a neighborhood containing words n to $n + N + margin$ is

$$s_i^{(n)} = \max_{j \in [n, n+N+margin]} (s_{ij}) \tag{1}$$

and we apply a threshold of similarity to 0.85 that performed the best based on experimentation. Now the total detection score in position n is the weighted mean

$$S^{(n)} = \frac{\sum_i w_i s_i^{(n)}}{\sum_i w_i} \tag{2}$$

A detection score is considered valid when it exceeds a threshold, expressing the algorithm's tolerance to names' variances.

4.6 Filters

Users often specify certain conditions under which their requests have to be executed. Our system uses such information to filter the collected data to create an answer. In terms of language processing, these filters are represented as adverbial clauses, whose main verb is connected to the root. To isolate such a clause, we apply Algorithm 1 starting from the clause's main verb. If this verb is omitted however, we search for a marker introducing a finite clause subordinate to the main sentence and examine nominal subjects of the root, that lie after this marker. We accept the adverbial clauses that are generated when an "is" or an "are" is added right after a nominal subject.

Simplified adverbial clauses can be abstractly represented as a triplet of the form "quantities-main verb-values". The verb part is the root verb of the clause. If absent, we consider the main verb to be a third-person-form of the verb "to be". Quantities stand for the part of the message where the quantities that specify the filter are declared and it is usually a noun phrase, formed by applying Algorithm 1 starting by nominal or clausal subjects of the main verb. Finally, values contain the part of the message where the exact filters are denoted and is a phrase generated by nominal and preposition objects, adjectival complements and prepositional modifiers of the main verb. Values are parsed like a logical function to specify the filter limits (e.g. *"more than 7 or less than 3"*). Negations of the main verb are also taken into account to invert the limits valuated.

4.7 Question Detection

Question detection is the binary problem of classifying a given sentence as a question or not. We expect a user would rarely use a question mark, so we build a deep syntactical approach. First, we mark as questions sentences that start with an auxiliary verb or a non-infinitive form of "be", followed by a nominal subject, an attributive or a compound. Furthermore, a question word that lies before the root, without being a direct object or a nominal subject, can also denote a question. Contrary to other works [23] that view the appearance of a question word (e.g. "who") as a feature for question detection, we demand that certain relations of these words with the sentence's root appear. Depending on the question word, accepted relations with the root can be an attributive ("what" or "who"), a nominal subject ("which") or an adverbial modifier ("where", "when", "how" or "why"). The above rules can reject sentences that have question-like form, such as "What you give is what you get", while accepting "What can you give".

4.8 Get a Quantity Vs Set a Quantity: A Syntax-Content Classifier

Deciding whether a message sets ("Set") or requests ("Get") a measurement is a problem of increased difficulty, as such sentences often have similar syntactic structure and content [20]. We build a complex algorithm that integrates syntactic and contextual information in multiple levels of analysis. We slightly benefit the "Get" category, as more common and less dangerous: a misclassification as "Set" could add an undesired measurement in the system's database, hence leading to future errors in data analysis.

Our algorithm starts with classifying as "Get" the messages that do not contain a main verb. Otherwise, we examine if the sentence has an interrogative form, using the algorithm described in Sect. 4.7. Questions are generally classified as "Get" messages, but formal requests are specially handled similarly to declarative and imperative sentences, examining the main verb and its subject. Sentences whose subject is not a pronoun are considered declarative that provide information, so we label them as "Set" messages. In case the subject is a pronoun, similarity scores between the main verb and a list of "Get" and "Set" verbs are computed, depending on whether the subject is a first- or second-person pronoun, then, we classify the message to the category with the maximum score, if that exceeds a threshold. In a different case, we look for open clauses and classify those sentences recursively. If no more clauses can be found and the message has not been classified yet, it is classified as "Get".

5 Evaluation and Results

The most common evaluating criterion for a chatbot is its ability to pass the Turing test [38] and "fool" a human judge [39]. However, more recent works [40] argue that giving the impression of being human is not a valid quality attribute, especially for task-oriented domain-specific chatbots, as Metis. Therefore, we follow the approach of [40] that evaluates the effectiveness and efficiency of the system along with users' satisfaction. As this work focuses mainly on handling the users' messages using NLP techniques, the above criteria are tested based on the language interface Metis provides. Throughout the examples, it can be seen that two are the main causes of possible failure, the lack of structure or correct syntax and SpaCy parser's misclassification.

5.1 Effectiveness

A chatbot's effectiveness reflects its analytical capabilities and therefore its evaluation consists of several interesting results from the point of computational linguistics.

First, the chatbot must interpret commands accurately. Our system uses three classifier models, trained on a corpus we collected, using messages from both experts of maritime communications and customers. We evaluate the accuracy of our models using 3-fold cross validation. Strong classifiers are able to achieve up to 85% accuracy, while the weak classifier can reach up to 100% accuracy on training data, despite its 65% accuracy on validation sets. The combined approach outperforms any of the three classifiers alone, achieving 98% on training data and over 90% on validation sets.

Secondly, the chatbot has to extract the requested parameters to execute a task. Regarding the type of the task, setting a value for a quantity or requesting a measurement (Sect. 4.7), we evaluate question detection on 1686 question-answer pairs from [41], where our algorithm was able to detect correctly 1523 questions (90.3%) when question mark is removed. Moreover, only 6 of 1686 non-questions were detected as questions (0.036%), resulting to 95% total accuracy on question detection. We noticed that questions that were misclassified as non-questions mostly had the form of "Garbage is what", while for others, our algorithm fails to handle SpaCy's parsing output. This is more common when named entities are involved.

Deeper into parameter extraction, a big challenge is date extraction, where our approach successfully identifies and parses 221 of 300 (73.7%) date formats of a collected dataset. Another challenge is the parsing of adverbial clauses to compute filters' parameters. An example could be *"Get last week's speed measurements when wind speed is more than 7 or less than 3 beauforts"*. The adverbial clause can be detected as *"wind speed is more than 7 or less than 3 beauforts"*, where *"wind speed"* is the quantities part, *"is"* the clause's main verb and *"more than 7 or less than 3 beauforts"* is the values part. The last one is analyzed further to extract the filter limits as the function to be applied on the measurements, in a form similar to "OR(>7,<3)".

Ease of use and answering to specific questions are other measures of a chatbot's effectiveness. Metis interacts in natural language and can handle a wide variety of message topics and formats, thus making it easy to use. In addition, as a task-oriented chatbot, it answers to standard questions, in terms of messages categories, in a way that it is a useful product that does not force newcomers to learn a new technology.

Lastly, the chatbot has to contain breadth of knowledge and to be flexible in interpreting it. In Metis, expert knowledge is embedded both in messages' analysis (e.g. abbreviations' handling, weights' initialization during quantity detection etc.) and in task execution. Table 1 summarizes evaluation where a dataset was available.

Table 1. Evaluation results of algorithms where a dataset was available

Problem	Accuracy	Dataset
Message classification	90%	Validation Set
Question detection	95%	[41] Q-A Pairs Dataset
Date parsing	73.7%	Collected Dataset of 300 Date Formats

5.2 Efficiency

When challenging the efficiency of a task-oriented chatbot, we evaluate its robustness to unexpected input and noise and its ability to handle such cases in a way that preserves the information given. Metis is robust to such inputs due to the strong classifiers and a detailed, domain-specific message preprocessing.

The first step when a message enters our system is to be checked for spelling and then simplified. Here, an unstructured message like *"Speed knots are three MT VESSELNAME"*, which would probably aim to set the speed of moving tanker VESSELNAME to 3 knots, could confuse the checker to generate the sentence *"Speed knots are three metric tonnes VESSELNAME"*. Our system is robust to most of such errors, thanks to the multiple classification stages that follow.

Conjunction splitting is also a process that can be confused by the abstract or complex syntax and omitted punctuation. For instance, in the sentence *"Give me the position of vessel-name 1 and the speed of vessel-name 2 when weather was bad"* it is unclear whether the adverbial clause *"when weather was bad"* refers to each one of the requested quantities in conjunction. We consider that such clauses refer to both coordinated terms, so we obtain *"Give me the position of vessel-name 1 when weather was bad"* and *"Give me the speed of vessel-name 2 when weather was bad"*. General parser parameters are also shared between the terms of a conjunction, similar to clauses, ensuring that no loss of information occurs in intermediate processing stages.

5.3 Satisfaction

Users' satisfaction is an important yet subjective criterion that combines affect, behavior and accessibility [40]. Affect measures users' willingness to communicate with the chatbot. Metis makes tasks easier and more pleasant, as with a simple message, the user gains access to an immense volume of data that can be processed or visualized in seconds. In terms of behavior, Metis is socially aware and protect and respect users' privacy and personality, constructing responses that are invariant to the users' social context and keeping every conversation is personal. Lastly, considering accessibility, Metis uses classifiers to detect the users' intent and the meaning of their messages and it can respond back quickly (the whole NLP pipeline needs about 2 s to process an average message), encouraging the user to ask another question.

6 Conclusion

Maritime is a field that could extremely benefit from an intelligent conversational agent able to embed efficient algorithms and experts' knowledge to understand and execute complex requests. We provide such a solution with the implementation of a personal assistant that communicates directly in natural language and is able to perform deep, multi-level analysis of messages to decide the topic and extract information about task execution. For these purposes, various algorithms were presented, while also showing their effectiveness with multiple tests and use-cases. An interesting future direction would be to examine the employment of Machine Learning classifiers to model parsing difficulties when handling unstructured or very complex messages during parameter extraction. We are also working on recognizing more relative dates' formats and types of filters. We believe the product we develop can practically facilitate maritime employers' and employees' everyday life, while preserving the simplicity of its usage.

References

1. Sameera, A.A.K., Woods, D.J.: Survey on chatbot design techniques in speech conversation systems. Int. J. Adv. Comput. Sci. Appl. **6**(7), 72–80 (2015)
2. Vtyurina, A., Fourney, A.: Exploring the role of conversational cues in guided task support with virtual assistants. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (2018)
3. Takagi, N., John, P., Noble, A., Björkroth, P., Brooks, B.: Vts-bot: using chatbots in smcp-based maritime communication. In: Proceedings of the Japanese Institute of Navigation JIN Conference, pp. 90–93 (2016)
4. Lasecki, W.S., Wesley, R., Nichols, J., Kulkarni, A., Allen, J.F., Bigham, J.P.: Chorus: a crowd-powered conversational assistant. In: Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology, pp. 151–162 (2013)
5. Feng, D., Shaw, E., Kim, J., Hovy, E.: An intelligent discussion-bot for answering student queries in threaded discussions. In: Proceedings of the 11th International Conference on Intelligent User Interfaces, pp. 171–177 (2006)
6. Hallili, A.: Toward an ontology-based chatbot endowed with natural language processing and generation. In: 26th European Summer School in Logic, Language & Information (2014)
7. Wilcox, B.: Chatscript, github repository (2018). https://github.com/bwilcox-1234/ChatScript
8. Li, J., Monroe, W., Ritter, A., Jurafsky, D., Galley, M., Gao, J.: Deep reinforcement learning for dialogue generation. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 1192–1202 (2016)
9. Augello, A., Saccone, G., Gaglio, S., Pilato, G.: Humorist bot: Bringing computational humour in a chat-bot system. In: Proceedings of International Conference on Complex, Intelligent and Software Intensive Systems, pp. 703–708 (2008)
10. Vinyals, O., Le, Q.V.: A neural conversational model. In: Proceedings of the 31st International Conference on Machine Learning (2015)
11. Higashinaka, R., et al.: Towards an open-domain conversational system fully based on natural language processing. In: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pp. 928–939 (2014)

12. Nigam, K., McCallum, A.K., Thrun, S., Mitchell, T.: Text classification from labeled and unlabeled documents using em. Mach. Learn. **39**(2–3), pp. 103–134, May 2000
13. Khan, A., Baharudin, B., Lee, L.H., Khan, K., Tronoh, U.T.P.: A review of machine learning algorithms for text-documents classification. J. Adv. Inf. Technol. **1**, 4–20 (2010)
14. Whitelaw, C., Hutchinson, B., Chung, G., Ellis, G.: Using the web for language independent spellchecking and autocorrection. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pp. 890–899 (2009)
15. Desai, N., Narvekar, M.: Normalization of noisy text data. Procedia Comput. Sci. **45**, 127–132 (2015). International Conference on Advanced Computing Technologies and Applications (ICACTA)
16. Chandrasekar, R., Doran, C., Bangalore, S.: Motivations and methods for text simplification. In: Proceedings of the 16th International Conference on Computational Linguistics, pp. 1041–1044 (1996)
17. Siddharthan, A.: Text simplification using typed dependencies: a comparision of the robustness of different generation strategies. In: Proceedings of the 13th European Workshop on Natural Language Generation, pp. 2–11 (2011)
18. Feblowitz, D., Kauchak, D.: Sentence simplification as tree transduction. In: Proceedings of the 2nd Workshop on Predicting and Improving Text Readability for Target Reader Populations, pp. 1–10 (2013)
19. Bingel, J., Søgaard, A.: Text simplification as tree labeling. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pp. 337–343 (2016)
20. Quinn, K., Zaiane, O.: Identifying questions & requests in conversation. In: Proceedings of the 2014 International C* Conference on Computer Science & Software Engineering - C3S2E, vol. 14 (2014)
21. Li, B., Si, X., Lyu, M.R., King, I., Chang, E.Y.: Question identification on twitter. In: Proceedings of the 20th ACM International Conference on Information and Knowledge Management, pp. 2477–2480 (2011)
22. Kwong, H., Yorke-Smith, N.: Detection of imperative and declarative question-answer pairs in email conversations. In: Proceedings of the 21st International Joint Conference on Artificial Intelligence, pp. 1519–1524 (2009)
23. Zhao, Z., Mei, Q.: Questions about questions: an empirical analysis of information needs on twitter. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 1545–1556 (2013)
24. Chambers, N., Jurafsky, D.: Template-based information extraction without the templates. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies 1
25. Mooney, R.J., Bunescu, R.: Mining knowledge from text using information extraction. SIGKDD Explor. Newslett. **7**(1), 3–10 (2005)
26. Microsoft: Bot framework emulator, github repository (2018). https://github.com/Microsoft/BotFramework-Emulator
27. Wilbur, W.J., Kim, W., Xie, N.: Spelling correction in the pubmed search engine. Inf. Retrieval **9**(5), 543–564 (2006)
28. Wang, P., Berry, M.W., Yang, Y.: Mining longitudinal web queries: trends and patterns. J. Assoc. Inf. Sci. Technol. **54**(8), 743–758 (2003)
29. Rojo-Laurilla, M.A.: English for maritime purposes: communication apprehension and communicative competence among maritime students in the philippines. Reflections Eng. Lang. Teach. **6**(2), 39–58 (2018)

30. Apostol-Mates, R., Barbu, A.: Human error-the main factor in marine accidents. Sci. Bull. Naval Acad. **19**(2), 451–454 (2016)
31. Jones, K.S.: A statistical interpretation of term specificity and its application in retrieval. J. Documentation **60**(5), 493–502 (1972)
32. Cetintas, S., Si, L., Xin, Y.P., Zhang, D., Young Park, J.: Automatic text categorization of mathematical word problems. In: Proceedings of the Twenty-Second International FLAIRS Conference, pp. 27–32 (2009)
33. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
34. Marneffe, M.C.D., Manning, C.D.: Stanford Typed Dependencies Manual (2008)
35. Honnibal, M., Johnson, M.: An improved non-monotonic transition system for dependency parsing. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1373–1378 (2015)
36. Stelmach, J.: Natty date parser (2018). http://natty.joestelmach.com/
37. Salton, G.: Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer. Addison-Wesley Longman Publishing Co. Inc, Boston (1989)
38. Turing, A.M.: Computing machinery and intelligence. Mind **LIX**(236), 433–460 (1950)
39. Shawar, B.A., Atwell, E.: Different measurements metrics to evaluate a chatbot system. In: Proceedings of the Workshop on Bridging the Gap: Academic and Industrial Research in Dialog Technologies, pp. 89–96 (2007)
40. Radziwill, N.M., Benton, M.C.: Evaluating quality of chatbots and intelligent conversational agents. CoRR abs/1704.04579 (2017)
41. Noah, A., Smith, M.H., Hwa, R.: Question generation as a competitive undergraduate course project. In: Proceedings of the NSF Workshop on the Question Generation Shared Task and Evaluation Challenge (2008)

Knowledge Acquisition Using Computer Simulation of a Manufacturing System for Preventive Maintenance

Sławomir Kłos(✉)

Faculty of Mechanical Engineering, The University of Zielona Gora, Zielona Góra, Poland
s.klos@iizp.uz.zgora.pl

Abstract. Preventive maintenance is an important component of the 'Industry 4.0 Concept'. Modern industry requires intelligent, autonomous and reliable manufacturing systems. Should a manufacturing system fail, it should be able to reorganise itself and put into effect a production plan, based on a scenario of selected activities. The problem is just how to predict possible failures and prepare scenarios for the behaviour of a system. With regard to the preventive maintenance system, potential failures can be modelled and then, by using computer simulation, based on simulation experiments, a database of maintenance knowledge can be created. In the paper, the methodology for the acquisition of maintenance knowledge, using the computer simulation method, is proposed. The example, put forward as an illustration, was prepared using Tecnomatix Plant Simulation software.

Keywords: Preventive maintenance · Knowledge acquisition
Computer simulation · Industry 4.0

1 Introduction and Literature Review

The Industry 4.0 Concept (I4.0) assumes integration and the close co-operation of the technical resources of manufacturing systems with human resources. Simple operations will be undertaken with the co-operation of robots and machines which will interact with products in order to optimise production flow. The preventive maintenance (PM) system includes the systematic inspection, detection and correction of incipient failures either before they occur or before they develop into major defects. PM requires equipment maintenance to be undertaken regularly, in order to reduce the probability of failure and uses the knowledge database to discover the possible causes and results of failures. PM ensures that everything of value, within your organisation, receives consistent maintenance, in order to avoid unexpected breakdowns and costly disruption. Of course, the elimination of failure in a complex manufacturing system is not possible. Therefore, the integral component of the preventive maintenance system should support a decision-support system which, in cases of failure, would propose a scenario to reduce the consequences of failure to a minimum (Maintenance Prevention).

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 29–40, 2018.
https://doi.org/10.1007/978-3-319-99972-2_3

In order to study the eventual consequences of failure, a model of the manufacturing system should be created and the different cases of failures should be examined.

Predictive maintenance is the objective for many research studies. Mori and Fujishima introduce a remote monitoring and maintenance system for machine tool manufacturers [1]. Dong et al. present the monitoring and maintenance of equipment systems for mine safety. They establish a predictive maintenance system which is based on the technology of the Internet of Things in order to change the existing method for the maintenance of coal mining equipment [2]. In the literature, many decision-support tools have been proposed for effective maintenance operations; these are based on the computer-simulation method. Ni and Jin propose mathematical algorithms and simulation tools in order to identify data-driven, short-term, throughput bottlenecks, the prediction of windows of opportunity for maintenance, the prioritisation of maintenance tasks, the joint production-scheduling and maintenance-scheduling of systems and the management of maintenance staff [3]. Roux et al. recommend that tools be combined in order to collaborate, thus optimising multi-component preventive-maintenance problems. The structure of the maintenance-production system is modelled using a combination of timed, petri-nets and Parallel, Discrete-Event System-Specification models and is implemented in a virtual-laboratory environment [4]. Boschian et al. compare two strategies for operating a production system which is composed of two machines working in parallel and a downstream inventory supplying an assembly line. They developed a simulation model for each strategy so as to be able to compare them and to simultaneously determine the timing of preventive maintenance for each machine considering the total average cost per time unit as the performance criterion [5]. Rezg et al. (2005) consider a periodical maintenance policy, one which is based on the age of the equipment. Their approach was based on simulation and experimental design, in order to determine the optimal age to undertake preventive maintenance actions, as well as the optimal size of the buffer inventory and thus, minimise total operating costs [6]. Karim et al. proposed propose a concept for knowledge discovery in maintenance, focussing on Big Data and analytics. The concept focusses on the new knowledge discovery in maintenance [7]. Wan et al. proposed developing and evaluating the prototype of collaborative, maintenance planning system using, as an example, a machine tool, which indicated that significant improvement could be achieved in the content of management technology, such as computer aided engineering, product data and life-cycle management and enterprise resource planning systems, as well as in managing machine tool maintenance and service information including dynamic and unstructured knowledge [8]. Computer simulation can be used to build complex manufacturing systems, analyse the effectiveness of preventive or corrective maintenance and improve overall, operational reliability, the utilisation of manufacturing resources and the productivity of manufacturing systems [9]. Frazzon et al. propose a simulation model to evaluate the improvements in performance provided by integrated intelligent-maintenance systems and spare-part supply chains [10].

The main motivation of the contribution was to propose the mythology for knowledge acquisition in area of production maintenance using simulation methods. In the paper, a concept for a knowledge acquisition system, based on simulation models, is proposed for supporting preventive maintenance in discrete manufacturing systems. The simulation models of manufacturing systems are prepared using Tecnomatix Plan

Simulation Software. The following research problem is taken into consideration in the paper: *Given is a simulation model of a discrete manufacturing system; the problem is: how to develop a knowledge database, based on the analysis of simulation experiments of predictive maintenance?* In the next chapter, the methodology for the acquisition of data, based on simulation experiments, is proposed.

2 The Methodology for Acquiring Knowledge Using Computer Simulation

The acquisition of the knowledge of manufacturing systems, using computer simulation, requires the proper definition of the manufacturing resources and specification of the main objectives for using that knowledge. In Fig. 1, the methodology for the acquisition of knowledge, for the preventive maintenance of a discrete manufacturing system, is proposed according to *waterfall software development methodology.*

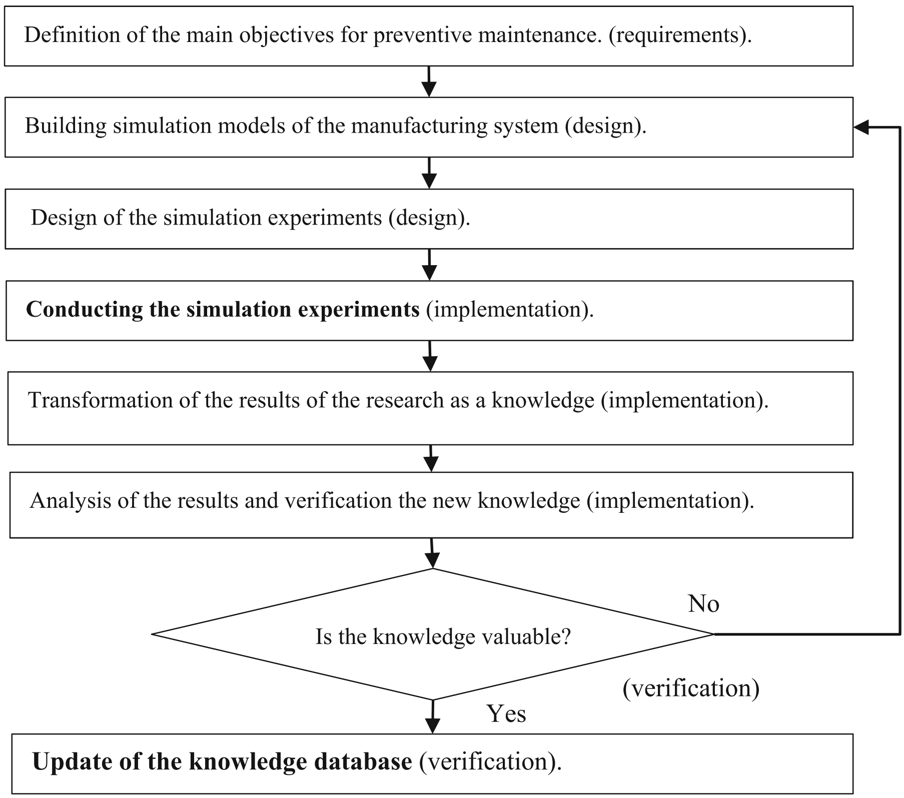


Fig. 1. Methodology for the acquisition of knowledge for preventive maintenance

The steps denoted using bold could be realized automatically. In the first step, a definition of the primary objectives for preventive maintenance is required. For example, the quantitative goal of preventive maintenance can be formulated as obtaining the highest throughput of the system using the smallest number of maintenance personnel. For the main objectives defined, the simulation model of the system should be created and the input and output variables for the simulation experiments should be determined. In the next step, the simulation experiments should be conducted and should include simulation times, numbers of observations, *etc*. The results of the simulation experiments should be generated as tables or charts in order to transform them into a knowledge format such as decision rules, logical or mathematical relations, *etc*. In the last step, new knowledge, determined by the simulation experiments should be verified as to its suitability for practical applications. If the knowledge proved useful, it could then be added to the knowledge database; if the knowledge did not to prove useful, then new simulation experiments should be prepared. In the next chapter, the simulation model of the discrete manufacturing system will be described. The main limitation of the proposed methodology is effort of building of simulation models.

3 The Simulation Model of the Manufacturing System

Simulation models of the parallel-serial manufacturing system include the structure of machines and buffers, the processing and setup times, batch sizes, dispatching rules and the availability of manufacturing resources [11]. The models are created using Tecnomatix Plant Simulation Software v. 12. It has been assumed that the system is fully automated but machine downtime requires the intervention of technical services. The first model of the system S1 is presented in the Fig. 2.

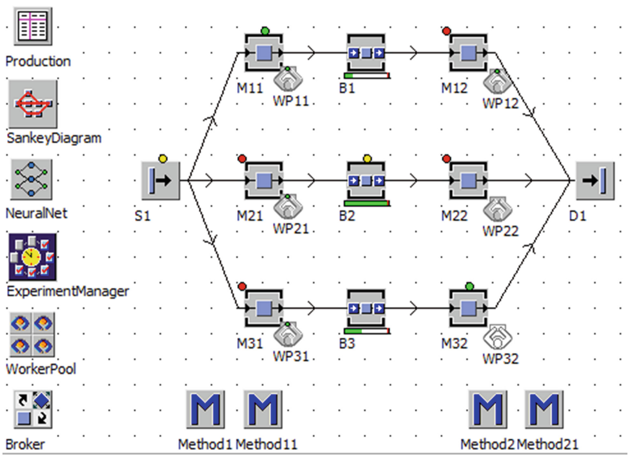


Fig. 2. The **S1** simulation model of the parallel serial manufacturing system

The model of the manufacturing system includes 6 machines, or manufacturing resources, which perform 2 technological operations. Machines M11, M21, M31 are identical and complete the first operation and machines M12, M22, M32 are also identical and complete the second operation. Between each pair of different machines, buffers B1, B2, B3 are allocated. The processing and setup times are based on a lognormal distribution. A lognormal distribution is a continuous distribution in which a random number has a natural logarithm that corresponds to a normal distribution. The realisations are non-negative, real numbers. The density of the lognormal distribution Lognor (σ, μ) is calculated as follows [12]:

$$f(x) = \frac{1}{\sigma_0 x\sqrt{2\pi}} \cdot \exp\left[\frac{-\ln(x-\mu_0)^2}{2\sigma_0^2}\right] \tag{1}$$

where σ and μ are respectively mean and standard deviations and are defined as follows:

$$\mu = \exp\left[\mu_0 + \frac{\sigma_0}{2}\right] \tag{2}$$

$$\sigma^2 = \exp(2\mu_0 + \sigma_0^2) \cdot (\exp(\sigma_0^2) - 1) \tag{3}$$

The maximum of the density function is defined as:

$$\exp(\mu_0 - \sigma_0^2) \tag{4}$$

The operation times are presented in the Table 1.

Table 1. The matrix of operation and setup times

Manufacturing resources	Operation times		Setup times	
	σ^2	μ	σ^2	μ
M11	480	20	840	200
M21	480	20	840	200
M31	480	20	840	200
M12	480	20	840	400
M22	480	20	840	400
M32	480	20	840	400

In the system, 4 products (A, B, C, D) are manufactured, based on the following sequence of production batches: 100 A, 300 B, 80 C, 120 D, 60 A, 200 B, 150 C, 80 D.

Four different products are manufactured; any change of batch requires additional setup times to be considered. For the processing of values and setup times, a set of 100 experiments for the random capacities of the buffers is generated.

For the model of manufacturing system presented here, a set of simulation experiments is designed and readied. The input values of the simulation are different allocations of buffer capacity -from 1 to 5- and availability of manufacturing resources - from 70%–90%. The values of input variables are presented in Table 2.

Table 2. The input values of simulation experiments

Exp	B1	B2	B3	M11	M21	M31	M12	M22	M32
01	1	1	1	70	70	70	70	70	70
02	1	1	1	90	90	90	90	90	90
03	1	5	2	70	81	72	76	74	84
04	2	5	2	78	82	79	87	85	72
05	2	4	5	87	79	87	74	89	72
06	1	5	5	75	87	72	78	72	70
07	1	4	2	82	78	86	79	76	76
08	5	4	2	86	87	84	87	89	81
09	3	5	1	73	72	87	77	75	85
10	2	5	4	72	86	71	81	86	78
11	4	4	2	72	73	74	82	72	73
12	3	5	4	83	84	73	74	73	86
13	2	4	2	80	88	73	71	77	82
14	5	1	5	83	86	74	74	79	73
15	2	5	3	76	83	82	84	80	71
16	5	3	2	73	89	73	75	82	76
17	3	2	2	79	79	83	75	70	74
18	3	1	4	70	81	82	87	80	86
19	2	4	4	87	75	79	83	73	73
20	3	5	5	77	76	79	79	76	72
21	4	5	4	77	88	89	85	79	87
22	2	1	5	85	77	71	84	90	89
23	5	5	5	70	70	70	70	70	70
24	5	5	5	90	90	90	90	90	90

The simulation experiments are conducted for different numbers of maintenance workers. The results of the output of the simulation experiments are analysed, along with the throughput per hour of the system and the average lifespan of the products, indicating the direct impact on the work in progress. The average lifespan of products is indicative of the average period of time that the products remain in the system. The input values of the simulation experiments are generated randomly, using the Tecnomatix Plant Simulation random generator. The first two experiments (Exp 01 and Exp 02) include the smallest value of buffer capacities (1) relating to the smallest and greatest values of the availability of resources. The last two experiments (Exp 23 and Exp 24) include the greatest values of buffer capacities relating to the smallest and greatest values of the availability of resources. In the next chapter the results of the simulation experiments are presented.

4 The Results of the Simulation Experiments

Each simulation experiment was conducted for a period of 240 h and repeated 3 times, making 3 observations. In Figs. 3 and 4, the results of the simulation experiments for 3 maintenance workers are presented.

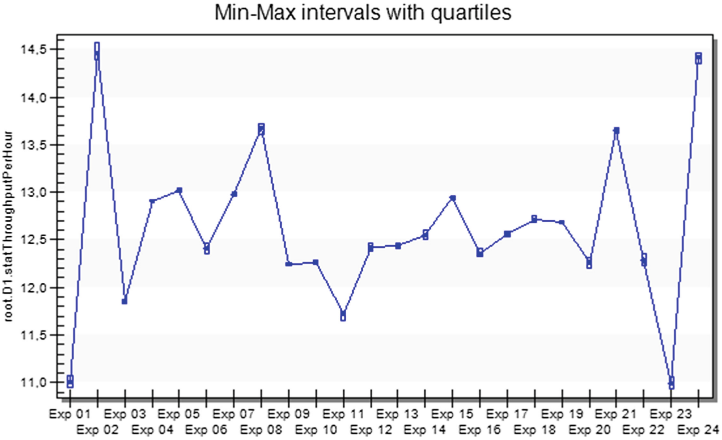


Fig. 3. The throughput of the system for model S1 and 3 maintenance workers

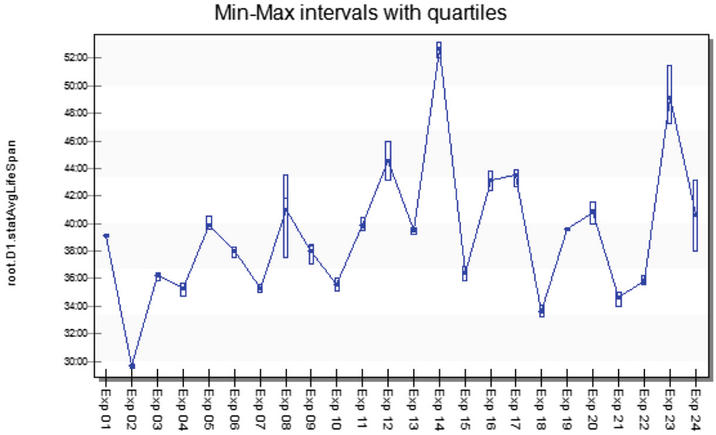


Fig. 4. The average lifespan of products for model S1 and 3 maintenance workers

The greatest throughput value and the smallest average lifespan value are obtained for experiment Exp 02. The smallest throughput value is obtained for Exp 23 and the greatest lifespan for a product is obtained for Exp 14. The simulation experiments are conducted for different number of maintenance workers (1–4). The results of the experiments are presented in Figs. 5 and 6, that is, throughput and average lifespan respectively.

The results indicate, as can be seen in the charts in Figs. 5 and 6, that the amount of maintenance work impacts significantly on the number of maintenance workers in the throughput of the system, that is, as a proportional relationship. If the number of maintenance workers is greater than 4, the throughput of the system does not change.

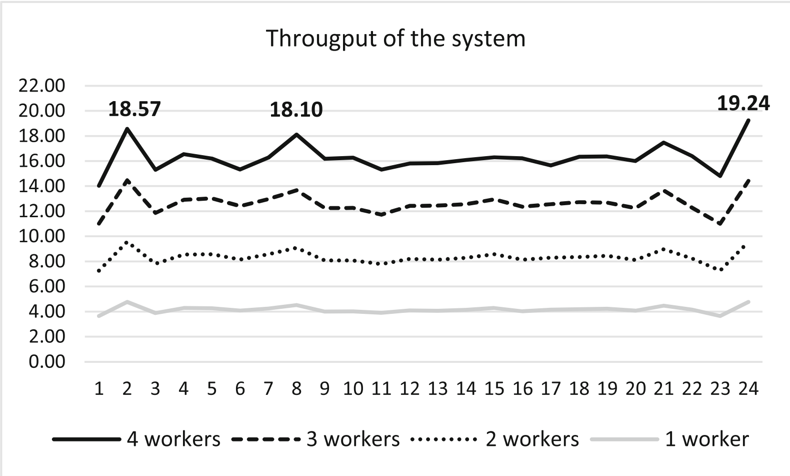


Fig. 5. The throughput of the system for model S1, with a different number of maintenance workers

Fig. 6. The average lifespan of products for model S1, with a different number of maintenance workers

The situation is different for average lifespan, that is, for *work-in-progress*. The greatest lifespan is obtained for a sole maintenance worker but for the variants with 3 and 4 maintenance workers, the lifespan values change differently. Generally, for experiments Exp 02, Exp 18 and Exp 23, the smallest, average lifespan value is obtained by the variant with 4 maintenance workers.

5 The Acquisition of Knowledge

Based on the simulation experiments conducted, the following new knowledge about a manufacturing system can be formulated as follows:

1. The number of maintenance workers impacts, proportionally, on the system's throughput.
2. For the system with 6 resources, the maximum number of maintenance workers is 4; a greater number has no impact, either on the throughput or on the average lifespan of the system.
3. The average lifespan of products for the number of maintenance workers greater than 2, does change but not proportionally.

If the new knowledge is not sufficient, then, by using the proposed methodology for the acquisition of knowledge, a new model can be created for the system (S2). The new model is presented in the Fig. 7.

The new model for the system includes dispatching rules on buffer exits and manufacturing resources; this means, for example, that machine M11 can send products to buffer B1 or to buffer B2, q.v. the Round Robin dispatching rule. For model S2, the same simulation experiments are conducted. The results of the simulation experiments are presented in Figs. 8 and 9. The implementation of dispatching rules results in an increase in the average throughput of the system.

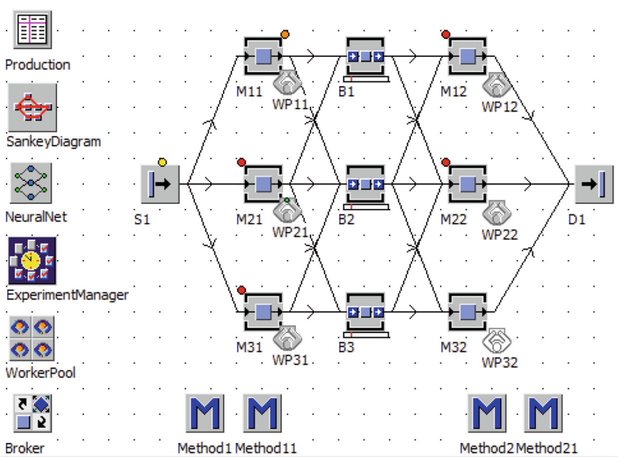


Fig. 7. The simulation model S2 of a parallel, serial, manufacturing system with dispatching rules

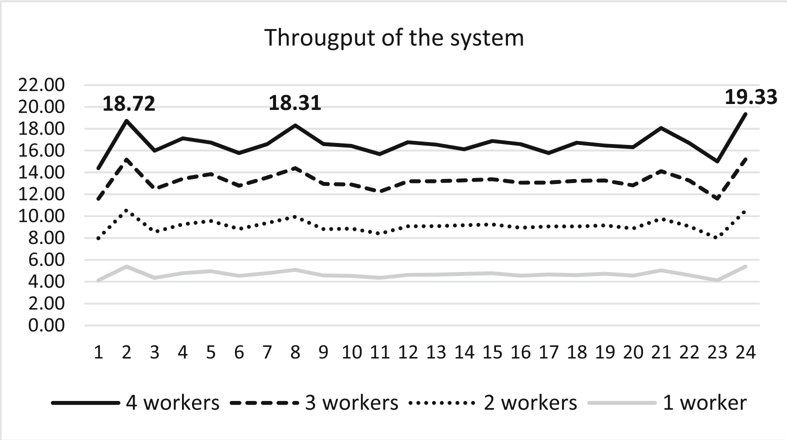


Fig. 8. The throughput of the system for model S2, with a different number of maintenance workers

Fig. 9. The average lifespan of products for model S2, with a different number of maintenance workers

The average lifespan for model S2 shows greater variability, especially for the variant with 4 maintenance workers. For model S2, with dispatching rules, it is possible to obtain greater throughput values than it is with model S1 using the same number of research workers. The percentage differences between the throughputs of models S1 and S2 are presented in Fig. 10. The percentage differences in the effectiveness of the system are greater than 14%, for the variant with the sole maintenance worker and about 6% (*q.v.* Exp 12) for the variant with 4 maintenance workers.

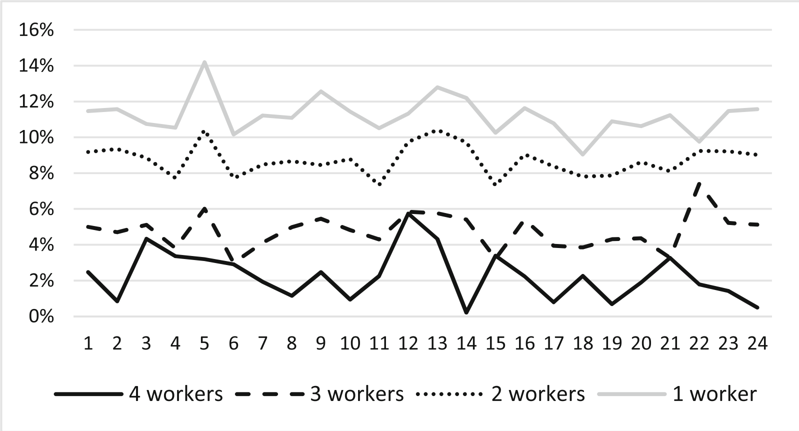


Fig. 10. The percentage difference between the throughput of models S2 and S1

Based on the new, manufacturing system model and on the results of the experiments conducted, the following, new knowledge may be formulated:

1. Changing the structure of a system, by implementing dispatching rules, increases the throughput of the system.
2. The average lifespan of products shows greater variability, especially for those variants with more than 1 maintenance worker.
3. Implementing dispatching rules could result in an increase in the average lifespan of work-in-progress products, for the variant with 4 maintenance worker; q.v. - as examples- Exp 05 and Exp 06.

In the next chapter, the final conclusions- as well as the directions for further research- will be presented.

6 Conclusions

In the paper, a methodology for the acquisition of knowledge, based on the computer simulation of a manufacturing system is proposed. The process for acquiring knowledge is presented on the basis of a parallel, serial, fully automated manufacturing system with different numbers of maintenance workers. The simulation experiments are conducted for two models of a manufacturing system, the first without having implemented dispatching rules, while the second had implemented dispatching rules, thus giving different structures for the flow of material. On the basis of the simulation experiments, new knowledge about the system is generated. This knowledge can be used in order to effectively manage maintenance personnel and thus, focus on increasing the system's throughput, or reducing the work-in-progress. On the basis of the results of the simulation experiments, a system of preventive maintenance could be proposed that would guarantee a certain level of effectiveness within a manufacturing system; any new knowledge could be recorded in the form of decision rules, for example:

If *Number_of_workers* = 4 then *Throughput* ∈ (14,39, 19,33);
If *Number_of_workers = 1 then Avg_lifespan* >57;
If *Number_of_workers = 2 and Variant = S2 then Throughput = Throughput*1,08.*

In further research, a tool, for the acquisition and recording of automated knowledge, will be designed and more indicators (not only throughput and lifespan) will be taken into account.

References

1. Mori, M., Fujishima, M.: Remote monitoring and maintenance system for CNC machine tools. Procedia CIRP **12**, 7–12 (2013)
2. Dong, L., Mingyue, R., Guoying, M.: Application of internet of things technology on predictive maintenance system of coal equipment. In: 13th Global Congress on Manufacturing and Management, GCMM, Procedia Engineering, vol. 174, pp. 885–889 (2017)
3. Ni, J., Jin, X.: Decision support systems for effective maintenance operations. CIRP Ann. Manufact. Technol. **61**, 411–414 (2012)
4. Roux, O., Duvivier, D., Quesnel, G., Ramat, E.: Optimization of preventive maintenance through a combined maintenance – production simulation model. Int. J. Prod. Econ. **143**(1), 3–12 (2013)
5. Boschian, V., Rezg, N., Chelbi, A.: Contribution of simulation to the optimization of maintenance strategies for a randomly failing production system. Eur. J. Oper. Res. **197**, 1142–1149 (2009)
6. Rezg, N., Chelbi, A., Xiaolan, X.: Modeling and optimizing a joint buffer inventory and preventive maintenance strategy for a randomly failing production unit: analytical and simulation approaches. Int. J. Comput. Integr. Manufact. **18**(2–3), 225–235 (2005)
7. Karim, R., Westerberg, J., Galar, D., Kumar, U.: Maintenance analytics – the new know in maintenance. IFAC-PapersOnLine **49**(28), 214–219 (2016)
8. Wan, S., Li, D., Gao, J., Roy, R., Tong, Y.: Process and knowledge management in a collaborative maintenance planning system for high value machine tools. Comput. Ind. **84**, 14–24 (2017)
9. Negahban, A., Smith, J.S.: Simulation for manufacturing system design and operation: literature review and analysis. J. Manuf. Syst. **33**, 241–261 (2014)
10. Frazzon, E.M., Silva, L.S., Pires, M.C.: Simulation-based performance evaluation of a concept for integrating intelligent maintenance systems and spare parts supply chains. IFAC-PapersOnLine **49**(12), 1074–1079 (2016)
11. Jasarevic, S., Brdarevic, S., Imamovic, M., Diering, M.: Standpoint of the top management about the effects of introduced quality system and continuation af activities of its improvement. Int. J. Qual. Res. **9**(2), 209–230 (2015)
12. Tecnomatix Plant Simulation version 12.2.0 on-line documentation

Information Systems: Special Session on e-Health and Special Session on Digital Transformation

Rate Your Physician: Findings from a Lithuanian Physician Rating Website

Frederik S. Bäumer, Joschka Kersting(✉), Vytautas Kuršelis, and Michaela Geierhos

Paderborn University, Paderborn, Germany
{fbaeumer,jkers,kurselis,geierhos}@mail.upb.de

Abstract. Physician review websites are known around the world. Patients review the subjectively experienced quality of medical services supplied to them and publish an overall rating on the Internet, where quantitative grades and qualitative texts come together. On the one hand, these new possibilities reduce the imbalance of power between health care providers and patients, but on the other hand, they can also damage the usually very intimate relationship between health care providers and patients. Review websites must meet these requirements with a high level of responsibility and service quality. In this paper, we look at the situation in Lithuania: Especially, we are interested in the available possibilities of evaluation and interaction, and the quality of a particular review website measured against the available data. We thereby identify quality weaknesses and lay the foundation for future research.

Keywords: Lithuanian physician review websites · Medical service ratings

1 Introduction

Review websites for Health Care Providers (HCP) such as physicians are a well-known nowadays [4–7, 9–12, 25, 26, 29]. So-called Physician Review Websites (PRWs) receive a lot of attention because they are online available to patients in many countries [29]. For example, there is the American website `Ratemds.com`[1], the German `Jameda.de`[2] or the Lithuanian "`Pincetas.lt`"[3]. They all have in common that they are well-frequented and record a high user interaction. PRWs also cover the large number of HCPs and attract a great deal of media attention [24], which is caused by HCPs, who do not want to be reviewed online [1]. This raises legal questions [2]. However, PRWs are not only seen negatively by HCPs because they use them for advertising purposes [20] although this is of questionable legality [8]. The high level of user interaction distinguishes PRWs from other medical information sources on the Web such as health care news (e.g. "`Sveikata.lt`"), health related products (e.g. "`Hiperfarma.lt`") or health communities (e.g. `Pasveik.lt/lt/forumas/`) [27]. Actually, the online-based reviewing of medical services seems to be the next

[1] http://www.ratemds.com, founded in 2004, ~ 1,700,000 HCPs, ~ 2,600,000 reviews.
[2] http://www.jameda.de, founded in 2007, ~ 275,000 HCPs, ~ 2,000,000 reviews.
[3] http://www.pincetas.lt, founded in 2006, ~ 60,000 HCPs, ~ 80,000 reviews.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 43–58, 2018.
https://doi.org/10.1007/978-3-319-99972-2_4

consequential step in the development of the Web 2.0, which already allows reviewing holiday resorts, films, products, etc. [20, 22, 23]. Additionally, multiple studies have been conducted on PRWs for many countries like Germany and the USA [6, 9, 10, 15]. When PRWs are analyzed, it is important to consider national influences: It must be pointed out that the PRWs can greatly differ in the way and quality of implementation. For example, this specifically applies to language, health system, rating scheme (e.g. star-rating vs. grade-based rating, qualitative vs. quantitative), the rating issues (e.g. friendliness, waiting time, parking, Wi-Fi) and the protection of patients' privacy. Until now, it has not been investigated in any Baltic country or any country in the Eastern part of the European Union. This situation is dissatisfying because local legal and social characteristics are reflected on these portals and are therefore worth investigating, especially, when considering the European Union's attempts for a unified single digital market across Europe [14]. Regarding the current legal cases of how to deal with PRWs' business model and the missing neutrality of PRWs between HCPs and patients, it is also unsatisfying not having investigated a PRW from the Eastern European Union [8]. Comparing different PRWs in detail makes it possible to identify good and bad idiosyncrasies on PRWs and to figure out and promote a better quality understanding. This is the primary purpose of this paper. Firstly, this study investigates one Lithuanian PRW by looking at the website data and by using natural language processing (NLP) techniques. This way, we evaluate the quality of the provided review data. Secondly, we briefly investigate online health information websites in Lithuania in order to underline the role of user-interaction which is unique to PRWs. Finally, we draw our conclusions from the used data, and discover future research topics such as a comparison of PRWs across Europe and the world.

The outline of this paper is as follows: Sect. 2 provides an insight into current research and even an explanation how the data acquisition and preprocessing steps were conducted. Section 3 gives an overview on our data set, and Sect. 4 presents our findings. In Sect. 5, we discuss our findings and compare them to international competitors. In Sect. 6, we conclude and give a brief outlook on future research.

2 Current State of Research

PRWs can significantly influence HCPs' success and they are useful and even decision-making for patients searching an HCP [10]. In 2013, one of six HCPs around the world was rated and the demand of PRWs is continuously growing [12]. As presented later, more than nine of ten reviews are positive. At the same time, any evidence for considerable doctor-bashing is denied, which is often discussed in the media [12].

The research so far only covers certain areas. There are some studies presenting data of PRWs from different countries [12, 15, 18]. Others analyze how German patients use PRWs [7]. Other studies investigate the patient's rating behavior by using German PRW data and investigate second-hand ratings [17], rating inconsistency by means of NLP [19], and cognitive bias in online reviews [28]. Other aspects like latent connectivity of HCPs (i.e. hidden connections among HCPs) are explored as well [4–6, 11]. Here, an important issue is privacy. Medical information is sensitive because, for example, users of PRWs can be identified because of their review contents [6].

Furthermore, others search for reasons how the ratings of physicians are achieved [21]. They therefore analyze reviewer statistics [29] and combine physicians' and patients' information to get better insights into the rating behavior [16]. But so far, there is no study collecting and using a data set from a North-Eastern European PRW like Lithuania. Furthermore, there is little research about Lithuanian websites in the medical sector. There is only one overview of Lithuanian medical information sources on the Internet [27]. In the following, we adopt the original table from 2014 with current websites but keep the general classification scheme (see Table 1).

Table 1. Lithuanian web sources for health care and lifestyle. Based on [27]

Section	Subsection	Online project
Informational	Overview & Library	Emedicina.lt
		Lmb.tl
	Health care news	Smlpc.lt
		Pasveik.lt
		Visivaistai.lt
		366.lt
		Farmapedia.lt
		Sam.lt
		Vlk.lt
	Health care news and products	Camelia.lt
		Eurovaistine.lt
		Zoopharma.lt
	Health related products	Hiperfarma.lt
		Hipereko.lt
		Hiperzoo.lt
		Ecohit.lt
		Ekomed.lt
		Nvaistine.lt
Participatory	Weak participation	Sveikaszmogus.lt
		Sveikata.lt
		Odontologija.com
		Mednews.lt
		Imunitetas.lt
	Consultations	Manosveikata.lt
		Konsultuokis.lt
		E-pacientas.lt
	Strong participation	Sveikasvaikas.lt
		Pincetas.lt

A low coverage of health-related websites in a comparatively small country like Lithuania seems quite normal. However, there is, as it can be seen in Table 1 a high number of different services. Since we deal with PRWs, we stick to the participatory section. For this purpose, our work is focused on a PRW categorized in Table 1 as "strong participation" in reviewing health care providers.

While [27] name "`Pincetas.lt`" as well-known PRW in Lithuania, we have serious doubts concerning the data quality and relevance of the website. One question is whether all HCPs really exist or whether there are errors and anomalies in the data. Furthermore, we are interested in the providers' specialties, their geographical distribution and average ratings. It will be interesting to view the reviews in combination with the corresponding provider. Until now, the distribution of reviews per HCP, the geographical distribution of reviews, etc. are unknown. For this reason, acquiring and analyzing data from "`Pincetas.lt`" will lead to new findings and provide a starting point for scholars dealing with similar research topics.

3 Descriptive Statistics

In order to present an example for and introduce to the data from PRWs, we provide a translated review from the Lithuanian website "`Pincetas.lt`" in Fig. 1. In general, reviews are divided into a quantitative and a qualitative part. The quantitative part includes the grades that can be awarded for categories such as friendliness and competence. They are shown as overall grades for all given ratings (summary). The qualitative part consists of an individual review text and a recommendation (green text color for recommendation).

Figure 1 shows an example for a Lithuanian physician review text translated into English. In addition to a timestamp (e.g. 2017-07-17, 11:55), the reviewer's IP address is also given (here shortened to keep anonymity), which is absolutely questioning concerning privacy issues. Moreover, HCPs can reply to a review. First, we have to acquire, build and preprocess a data set for our research purposes because no structured Lithuanian PRW data set is available right now.

3.1 Data Acquisition and Preprocessing

We use HCP profiles, medical institution profiles and free text reviews from the Lithuanian PRW "`Pincetas.lt`" as data source. The PRW was founded in 2006 and it is growing steadily, measured by the number of reviews submitted. The first review we acquired is dated on 14/07/2006. Overall, only 45 reviews were written in 2006, compared to 3,503 in 2007 and a total of 5,599 in 2008. This is a remarkable growth, which was repeated in 2015: While until 2015, between 6,000 and 8,000

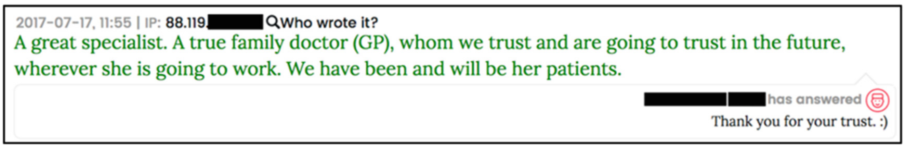


Fig. 1. Sample HCP review on `Pincetas.lt` (translated from Lithuanian)

reviews have been written per year, in 2016 it climbed up to 10,810 and in 2017 to 12,914 reviews. We acquired our data set between July and December 2017 by using a fully automated approach (similar to the procedure described in [4]). Therefore, a crawler was developed, which searched the PRW and gathered all listed HCPs in the website's index. Subsequently, the information was stored in a database and further processed, e.g. by tokenizing sentences, lemmatization, etc. Our data set contains 57,311 profiles of health care providers (e.g. names, licenses, workplaces), 2,983 medical institutions (e.g., area of expertise, address data), and 81,765 reviews for HCPs and medical institutions (e.g., review text, sentiment). Furthermore, the data set contains IP addresses, which can be assigned to the respective reviews. IP addresses of the reviewers who were not logged-in are published with the corresponding review text. These IP addresses were also used to identify the country, city, internet provider and geographic coordinates (longitudes, latitudes), which represent valuable additional information. Moreover, reviews without texts also exist, but are not separately listed. Nevertheless, preprocessing is still essential: Examples are the names of the HCPs, which do not exist as individual data fields (first names, surnames) but as a single string. Since separate first and last names are more suitable for further data processing, we split them during the preprocessing step. The same isolation technique is applied to medical licenses such as "*Vidaus ligų gydytojas Išduota: 2009-05-27, Nr. MPL-XXXXX, Galiojanti*" (Internal illnesses doctor Issued: May 27, 2009, No. MPL-XXXXX, Valid). Another example is the gender of a HCP. This information is not available on this PRW. Therefore, we have analyzed HCPs' first and last names, which provide information about gender in most cases. For this purpose, we used the internal word evidence such as typical Lithuanian first names (e.g. "*Birutė*") and last name suffixes (e.g. "*-ienė*"). On the one hand, each review is represented by its ID, HCP ID, timestamp, IP address, sentiment, review text (i.e. comment) and a reply by the doctor (if provided). On the other hand, each HCP is characterized by its ID, name, area of specialization, recommendation value, total number of ratings, licenses, workplaces, average ratings like how patients rate the diagnosis, question answering, etc. (grades from 1 to 5, where 5 is the best), and whether the HCPs offer an online appointment booking on their website.

3.2 Overview Over Health Care Providers, Medical Institutions and Reviews

In the following, we describe our data subsets (HCPs, medical institutions and reviews) individually, knowing well that they are strongly interlinked.

Health Care Providers. We acquired 57,311 HCP profiles, which cover 78 specialist areas. The three most represented areas of specialization are "*Bendrosios praktikos slauga*" (general nursing), "*Medicinos gydytojas*" (physician), "*Odontologija*" (odontology). About 5% of the acquired profiles did not specify any specialist area. It is surprising that the most frequent area is nursing. Since the specializations are accompanied by the corresponding licenses, it is also worth to review this information: In total, 87% of the provider's profiles contain information about licenses, whereof 85% have at least one valid license. Conversely, this means that 15% have no valid license and 13% of the providers have not provided any information about this.

Medical Institutions. Next to the provider's data, we also acquired 2,983 profiles of medical institutions including medical practices and hospitals. From the data set, it can be noted that HCPs are often assigned to more than one institution. Most of the providers with information about their workplace are assigned to only one medical institution (75.4%), followed by two (18.2%) and three institutions (5.8%). Only 1.6% of the HCPs are assigned to more than three institutions. This information is missing in 83.7% of acquired HCPs.

Reviews and Ratings. With 81,765 review texts, we have acquired quite a small amount of data compared to PRWs considered in the literature so far [6, 18]. This may be related to the different population sizes in Lithuania vs. the USA. The reviews are distributed in such a way that 6,750 HCPs (12%) are covered. However, this does not mean that these providers were not evaluated at all – in total 167,050 (quantitative) recommendations were provided. 9,124 HCPs (16%) have at least one rating. The average number of ratings for all HCPs with at least one rating is nine, with 48,187 HCPs (84%) having received no rating at all.

3.3 Identified Spam

Unlike the PRWs we have examined so far, "`Pincetas.lt`" uses only a few anti-spam techniques (e.g. captchas) and no anti-fake measures were identified. This is risky because spammers and fakers are interested in the data for several reasons: On the one hand, there are competing HCPs who maybe want to negatively review other HCPs or give positive ones to themselves. On the other hand, patients, who react emotionally to a perceived treatment, can review one or more HCPs several times. A third possible case could be that a PRW itself generates (fake) reviews. Due to this fact, we may have to exclude certain reviews from our investigation as they do not meet our quality standards. When the number of HCP's published reviews is compared to the given total amount of reviews on top of the page, we noticed mismatches. For example, one doctor is said to have 23 reviews while only 20 are visible. We assume the hiding of offensive and dubious reviews, especially as this sample HCP received very negative grades.

4 What Data Tells Us: Insights into Our Findings

In the following, we focus on the ratings, reviews, overall quality and evaluate the information about the reviewers. After that, we conduct a qualitative evaluation of the PRW and point out possible improvements.

4.1 HCPs in Detail

The highest number of qualitative reviews received by one HCP is 249 and 1,256 recommendations. These values are comparable to other European PRWs. HCPs' activity on PRWs is also comparable, there are HCPs who participate very actively on PRWs, comment on reviews, and maintain their profiles and there are some who seem to completely refuse this medium. It is mainly explained by the fact that the HCPs do not register themselves on these websites. Participation and approval of a HCP is not mandatory. This is of course a disadvantage because the overall data quality would increase through higher HCP participation. While HCPs' addresses are not provided on this PRW, the ones of workplaces like hospitals are made public (see Fig. 2). Since most HCPs are associated with workplaces such as hospitals and medical practices, we consider the workplaces as representative for a distribution of HCPs. As Fig. 2 shows, most HCPs are in Vilnius (capital city) and Kaunas (second largest city). Therefore, this is unsurprising because these are Lithuania's biggest cities [30]. However, rural areas seem to be underrepresented in comparison to bigger cities.

4.2 Ratings and Reviews

As already mentioned, a distinction must be made between ratings and reviews. It is possible to evaluate an HCP quantitatively and qualitatively, while the qualitative evaluation (review) is optional. For this reason, there are more quantitative ratings (167,050) than qualitative reviews (81,765). In other words, more than half of the submitted ratings are without review text. As stated before, we expect even some hidden reviews because 81,765 reviews are online, but arithmetically, there should be 83,926 reviews in total (i.e. sum of all review counters on the HCPs' profiles).

Eight dimensions can be quantitatively evaluated (see Table 2). Currently, all dimensions have to be rated. However, there are also HCP profiles in which not all dimensions are specified. This may be a mistake, or the rating system has been changed over the years. The peculiarity of the investigated PRW is that the reviewers do not assign a specific quantitative rating per dimension (such as 1 to 5 out of 5 stars) but select one of the predefined values, which are mapped to a floating-point grade scale from 1 to 5 by the PRW (5 is best). In addition to these dimensions, an indication has to be given, whether the HCP can be recommended or not. As already mentioned, the grades are interrogated by phrases. For example, for a very effective perceived treatment, a matching answer is "*Pilnai išgydė*" (cured completely) for the best rating (5). Furthermore, when it comes to the waiting period in a HCP's office, patients were asked to provide the waiting time. The conversion to a grade then provides e.g. zero minutes as grade 5 and 15 to 20 min as grade 3. For other users, there are only numerical grades visible on the HCP's profile. The profile coverage shown in Table 2 refers to how many of the 11,436 HCPs with at least one rating received a rating in this dimension. It turns out that all dimensions are rated well on average. This raises the question of whether the ratings may be different for the individual areas of specialization. And it is proven that within this data set, the ratings vary depending on the area of specialization: Table 3 shows the best and worst grades per rating dimension, excluding the area of specialization with less than ten ratings. In addition, only the primary specialization has been taken into account, as indicated by HCPs on their pages.

Further findings can be found in the written reviews. The 81,765 reviews consist of 74% positive and 26% negative reviews (based on recommendation). This fits existing research [24], according to which, despite all perceptions, PRWs are predominantly evaluated positively [12, 23]. In general, reviews tend to be either very positive or

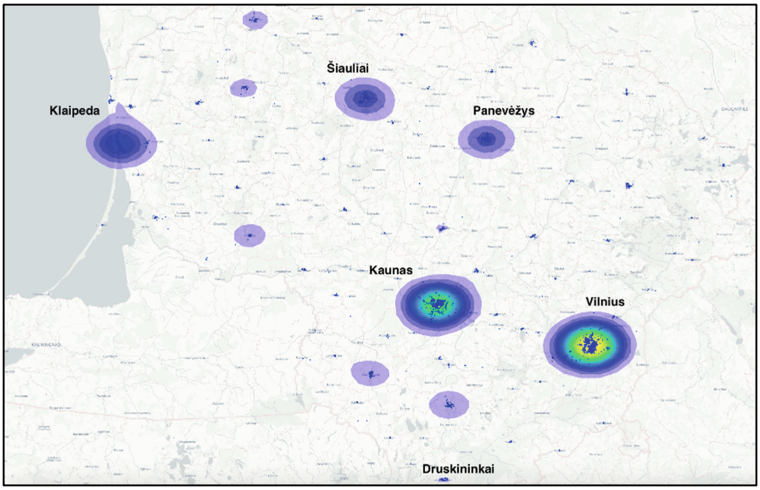


Fig. 2. Distribution of HCPs based on medical institutions

Table 2. Rating dimensions, grades and profile coverage

Rating dimension	Ø Grade	Coverage
Ar atsakė į visus Jums rūpimus klausimus? Were all your questions answered?	4.07 of 5	11,321 (**98.9%**)
Ar skyrė pakankamai laiko? Did you receive sufficient time for treatment?	4.02 of 5	11,308 (98.8%)
Kaip vertinate paslaugų kainas? How do you rate the service prices?	**4.42** of 5	11,242 (98.3%)
Kaip tiksliai diagnozavo Jūsų problemą? How accurately was your illness diagnosed?	4.13 of 5	11,236 (98.3%)
Kiek laiko laukėte, kol Jus priims? How long did you wait for the appointment?	**3.76** of 5	11,164 (97.6%)
Ar paskirtas gydymas buvo veiksmingas? Was the prescribed treatment effective?	4.03 of 5	11,106 (97.1%)
Ar po apsilankymo pasidomėjo kaip jaučiatės? Was there a follow-up contact?	3.96 of 5	8,879 (**77.6%**)
Ar atsilyginote gydytojui asmeniškai? Did you pay the doctor directly, i.e. extra?	No data acquired	

negative [22]. However negative reviews have a stronger impact than positive ones [31].

So far, we have not seen any PRW with a high response rate from the HCPs. There are, indeed, good reasons not to react [24]. It is the same in this analyzed PRW: Just 0.17% of the reviews received replies from the HCP. It is noticeable that these are not only reactions to negative reviews, but also words of gratitude. In this context, we have looked at the communication regarding the used sentiment words. Thus, the most

Table 3. Average ratings per area of specialization

Dimension	Ø Grades	Area of specialization
Answered questions	4.6 3.4	*Odontologija* (odontology) Vaikų pulmonologija (pediatric pulmonology)
Treatment time	4.6 3.4	*Odontologija* (odontology) *Neurologija* (neurology)
Service prices	4.9 3.3	*Vaikų hematologija* (children's Hematology) *Patologija* (pathology)
Diagnose accuracy	4.9 3.4	*Burnos higiena* (oral hygiene) *Vaikų neurologija* (child Neurology)
Waiting period	4.9 2.7	*Dantų technika* (dental technology) *Vaikų pulmonologija* (pediatric pulmonology)
Treatment efficiency	4.9 3.4	*Burnos higiena* (oral hygiene) *Neurologija* (neurology)
Follow-up contact	4.6 3.4	*Širdies chirurgija* (heart surgery) *Dermatovenerologija* (dermatovenerology)

common sentiment words (1- to 2-grams) in positive reviews are "*ačiū*" (thanks), "*puiki*" (great) and "*maloni*" (nice), while negative reviews are dominated by "(*labai*) *nemaloni*" (very unpleasant), "*nieko gero*" (nothing good), "*nerekomenduoju*" (not recommend). Here, it should be noted that a review can also just consist of one word, e.g. "*ačiū*" (thanks). The shortest reviews found on the PRW have a length of one character (e.g. ".", "+"), while the average in the whole data set is 188 characters. The longest review has 7,247 characters and is negative. However, on average, negative reviews (290 characters) are longer than positive ones (152 characters).

Because of the frequency analysis, we identified many duplicates in the database – even if looking for 6-gram phrases. The reasons for that can be different (e.g. spam). An example therefore is "*Sveiki, norėčiau pasidalinti savo nuomone*" ("Hello, I'd like to share my opinion"). A phrase that is found twice in the data set and written for the same HCP within four minutes. It can be assumed that it was an unintentional duplicate review. Other examples indeed suggest systematic spam activity. Since we have repeatedly recognized the use of phrases from other reviews in supposed spam reviews, we have developed a graph-based procedure for further analysis. We have divided all reviews into sentences and then identified those which are often used in reviews (Fig. 3). On one hand, this leads to sentences like "*buvo nemalonų*" (was unpleasant) being marked as frequent, which is not critical. On the other hand, reviews that are partially copied or even whole copies are detected. We split our 81,765 reviews into 167,123 sentences, of which 96.5% are unique and 3.5% are used in more than one review (full copies). We can only speculate about the motivation to mix different existing texts to create new reviews. In the case of review duplicates, we cannot exclude that patients copy existing reviews for the same HCP, when they share the same opinion. This would also explain why years have sometimes passed between such duplicates. Unfortunately, we have limited space in this study to describe this situation in more detail. However, one example for plagiarism is given in Fig. 4, which shows a case where all sentences are identical in both reviews. If we look at further details in

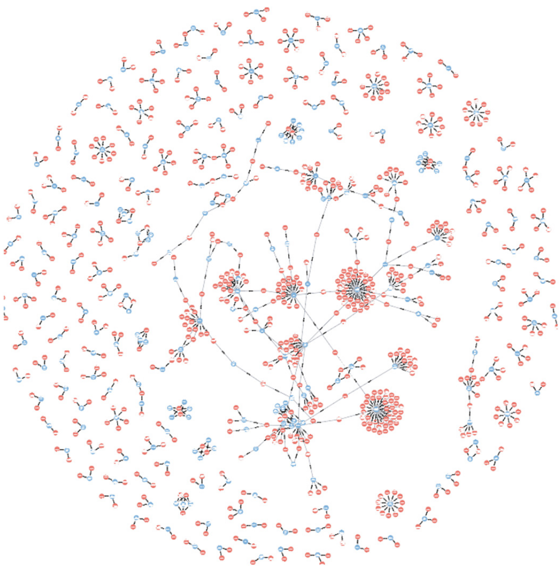


Fig. 3. Sentences and their appearance among different reviews

our data set, an anomaly can be assumed, since the reviews were written within five minutes for the same HCP from the same IP address.

It is more difficult to detect duplicates, in which marginal changes (in many cases numbers) are present (see Fig. 5). In this example, the HCP with ID 2497 has four ratings of this kind (three of them are visible in Fig. 5). While the first one was written in 2011, the other three reviews were written within three months in 2017. Since three of the four reviews can be assigned to the same IP address, this could be a patient who often visits the HCP and simply expresses the same opinion over and over again. However, this does not explain why so many other HCPs have identical review texts.

4.3 Insights into User's Review Behavior

In the following, we will take a look at the reviewers. What can we learn about those who give ratings and write reviews?

Reviewer Location. The acquired data set has a special feature that we have not had in similar data sets examined so far: Reviews are not anonymous but have a unique IP address. This is surprising, since previous work assumed that reviews in this domain deserve special privacy protection and PRWs are also interested in this protection [6]. At the same time, this protection of the reviewers ensured that it was not possible to analyze the reviewer's behavior. Now, this is possible with the clearly marked data. However, there are two aspects to consider: On the one hand, patients and reviewers do not have to appear in person [17]. This means that a review does not necessarily represent the opinion of the patient (third party representation). It also means that a reviewer does not even necessarily have to be the patient (fake reviews). On the other

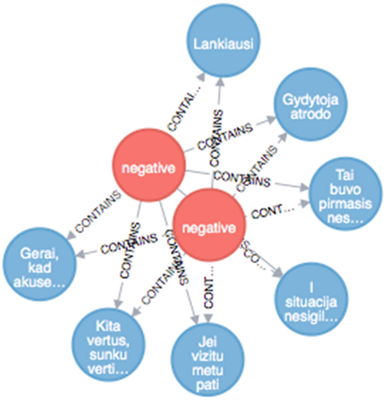


Fig. 4. Detected plagiarism

2497	Manau reikėtų pervertinti gydytojos kvalifikaciją. Pirmą kartą konsultacijos laukiau apie 30 min, registratūroje tuo metu slaugytojos
10199	Manau reikėtų pervertinti gydytojos kvalifikaciją. Pirmą kartą konsultacijos laukiau apie 30 min, registratūroje tuo metu slaugytojos
20135	Manau reikėtų pervertinti gydytojos kvalifikaciją. Pirmą kartą konsultacijos laukiau apie 30 min, registratūroje tuo metu slaugytojos
2751	Manau reikėtų pervertinti gydytojos kvalifikaciją. Pirmą kartą konsultacijos laukiau apie 45 min, registratūroje tuo metu slaugytojos
2497	Manau reikėtų pervertinti gydytojos kvalifikaciją. Pirmą kartą konsultacijos laukiau apie 45 min, registratūroje tuo metu slaugytojos
2497	Manau reikėtų pervertinti šitos gydytojos kvalifikaciją. Pirmą kartą konsultacijos laukiau apie 45 min, registratūroje tuo metu slaug
8274	Manau reikėtų pervertinti šitos gydytojos kvalifikaciją. Pirmą kartą konsultacijos laukiau apie 45 min, registratūroje tuo metu slaug

Fig. 5. Examples of duplicates with marginal changes

hand, reviewer can hide IP addresses (by using e.g. VPN, proxy servers) or IP addresses are used by several people (e.g. public Wi-Fi). This is possible in public areas such as medical institutions, stations of public transportation or catering establishments.

By taking into account the mentioned limitations, a few interesting findings can be derived: In total, ratings from 80 countries were submitted. The locations from all over Europe can be found in Fig. 6. Unsurprisingly, since "`Pincetas.lt`" is only available in Lithuanian, 94% of the reviews come from Lithuania (55.2% from Vilnius, 21.6% from Kaunas and 8.9% from Klaipėda). Another 1.3% come from the United Kingdom, 0.8% from the USA and 0.6% from Germany. This distribution of countries is in line with emigration movements[4] and therefore appears coherent. Furthermore, there is no significant difference in the distribution of positive and negative reviews with reference to the country of origin. While there were no significant numbers of reviews from Russia (0.2%, mostly from Moscow), still a few Russian reviews can be found. This is also remarkable because the registration process is in Lithuanian, too. More noticeable is the number of ratings given per IP address. Are there IP addresses that rate very often or that always rate only one provider? This could be useful for spam and fake review detection. On average, 1.6 reviews are assigned per IP address, while 74% of IP addresses are assigned to only one review, 16% are assigned to two, and 5% to three reviews. This number of reviews is considering that several doctors, nurses and

[4] See for more information: http://123.emn.lt/en/, accessed 16/01/2018.

pharmacists can be involved in a treatment as HCPs. Especially without any temporal information, this number can be judged as inconspicuous IP addresses with five or more reviews (3.4%) are more interesting when they are written within a defined period (e.g. month). There are several examples in the data set: An eye-catching example is an IP address (185.127.*.*) which wrote 198 reviews within June to December 2017 and thus evaluated 148 HCPs. On one single day, 21 reviews were written for different HCPs, but all of them, with one exception, were negative. It is also very noteworthy that the texts are very similar: Some of them are 1:1 plagiarism, some of them are copied texts with marginal changes. What is surprising here is the fact that the phrases are also used in reviews of others.

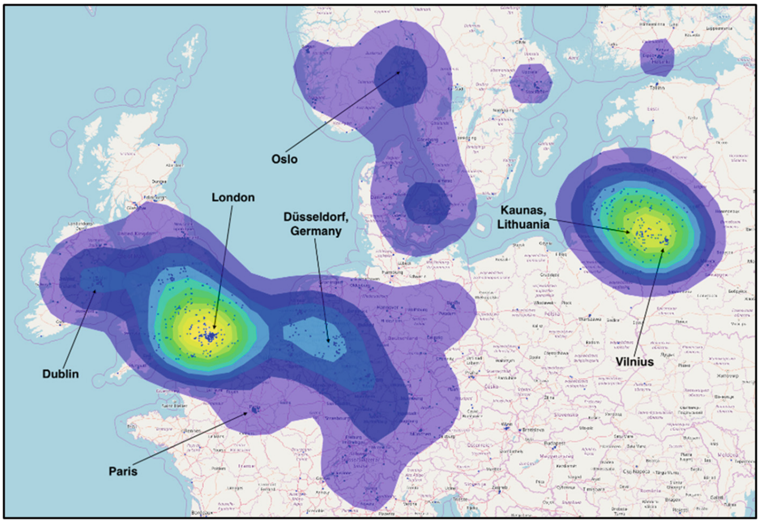


Fig. 6. European reviewer locations based on the provided IP addresses

5 Looking Beyond Own Data: A Discussion

The investigated PRW shows several differences compared to international competitors. This concerns the on-page functions on the one hand and the quality of provided information (in particular the review texts) on the other hand, e.g. protection mechanisms against spam.

On the positive side, it should be noted that the PRW reflects well the national characteristics. We have rarely seen PRWs listing licenses of HCPs, which is a favorable feature. It is particularly interesting to see which licenses have expired. This ensures high transparency (assuming that this data is also maintained). Showing IP addresses of reviewers is difficult to evaluate. On the one hand, this can prevent abuse;

on the other hand, this may harm the privacy of users. As the debate about cyberbullying[5] is strong in Lithuania [3, 13], the PRW meets national standards here. Whether it would be a better solution, for example, to use hash values or unique IDs, is at least to be considered. IP addresses allow user tracking across pages, which should be avoided due to privacy concerns. Assuming that reviews are usually written honestly, users face a serious lack of privacy. Nevertheless, showing an identifier of non-registered users demonstrates a fair-use policy to everyone. An interesting finding derived from the IP addresses is the geographical distribution of the reviews. While most reviews come from Lithuania, the others are mostly sent from common emigration destinations of Lithuanians. This is a plus for the data quality and a finding that cannot be drawn from other PRWs. In addition, the IP addresses gave us a clue to detect spam and fake reviews. Unfortunately, we found that the quality of this PRW is significantly lower than comparable PRWs. While "`Jameda.de`" has between 0 and 223 reviews per HCP, the Lithuanian PRW has between 0 and 249 reviews though being a considerably less-used service. It is interesting that at "`Jameda.de`", there are only 2% of reviews exclusively quantitative [18], while on this PRW there are roughly 50% quantitative. Moreover, the regarded PRW could be improved by making use of the true rating dimensions. That is, users rate HCPs according to textual expressions which are converted into grades. Giving the information that, for example, grade 3 is a waiting period of 15–30 min would be beneficial for all involved parties and more informative than numerical grades, especially due to various ways of understanding grades.

Furthermore, the function to link research and news articles related to a HCP is an interesting next step providing more information about the professionality of the HCPs. In comparison, "`Jameda.de`" provides an article feature while recommending topic-related articles, when viewing a HCP's profile. Furthermore, they provide detailed reviews including single grades given per review. Additionally, they provide business hours, addresses and so forth. A non-distinguishing function is the self-presentation on the German PRWs. HCPs can upload pictures of their practice, specify their treatment focus, education, etc. Here, the investigated Lithuanian PRW seems rather simple. If necessary, payed HCP pages and features could also be a business concept for "`Pincetas.lt`." However, "`Jameda.de`", is criticized for precisely this service because it is assumed that paying HCPs receive better grades. Moreover, on-page advertising doesn't make PRWs appear particularly trustworthy.

6 Conclusion and Future Work

This study investigates Lithuanian medical information websites and the largest Lithuanian PRW in particular. We chose Lithuania because it has a PRW that experiences a high usage as well as growth during the last years, as stated before. It became clear that Lithuania has a large number of information portals covering a wide range of information (Sect. 2). We have also shown that PRWs are a well-researched area and

[5] See https://cyberbullying.org, accessed: 22/01/2018.

that they play an important role for independent patient information. On the one hand, PRWs are navigators in the medical profession and on the other hand, they serve as an interaction platform for patients, which can reduce the imbalance of power between HCPs and patients. Unfortunately, this is precisely where we found out that this Lithuanian PRW still has some shortcomings.

The overall quality of the review texts is not satisfactory, because there are a lot of fake and spam reviews. Furthermore, it is not certain whether the reviewers are really patients, since no verification of the information takes place. Moreover, it is not possible for patients to understand how the overall grades of the HCPs are calculated, as the grades are not broken down. More transparency is needed on the analyzed PRW.

In sum, it can be said that the studied PRW is a valuable tool for patients to obtain and share information about HCPs. Here, HCPs get the chance to receive and comment on the concerns and feedback of patients. The offered features are comparable with other PRWs from Germany and the USA. However, since patients rely on the quality of information, there are still some improvements to be done. As we have opened the topic of international PRWs, we are keen on further studying these websites. As demonstrated by earlier research, PRWs are already influencing HCP's performance and as PRWs are a central mean for choosing a HCP, further research in this area will be productive. This study has investigated quality information on a Lithuanian PRW and made a brief comparison to prior experiences with German and American PRWs.

In the future, we want to extensively investigate PRWs from several countries in order to provide a broad-based comparison. We are interested in Eastern European, Central European, North American and Asian PRWs. We expect not only qualitative results from a comparison, but data-based findings gathered by NLP approaches.

References

1. Ärztezeitung,"jameda und co: Gericht stärkt Patientenposition bei Arzt-Bewertung" (2018). https://www.aerztezeitung.de/praxis_wirtschaft/recht/article/955200/jameda-co-gericht-staerkt-patientenposition-arzt-bewertung.html. Accessed 22 Jan 2018
2. Apotheke-Adhoc, "Von Jameda zur Konkurrenz geschickt" (2018). https://www.apotheke-adhoc.de/nachrichten/detail/apothekenpraxis/von-jameda-zur-konkurrenz-geschickt-bewertungsportale/. Accessed 22 Jan 2018
3. Baraldsnes, D.: The prevalence of cyberbullying and the views of 5-12 grade pupils and teachers on cyberbullying prevention in lithuanian schools. Univ. J. Educ. Res. **3**(12), 949–959 (2015)
4. Bäumer, F.S., Dollmann, M., Geierhos, M.: Find a physician by matching medical needs described in your own words. Procedia Comput. Sci. **63**, 417–424 (2015)
5. Bäumer, F.S., Geierhos, M., Schulze, S.: A system for uncovering latent connectivity of health care providers in online reviews. In: Proceedings of the 21st ICIST, Druskininkai, Lithuania, vol. 538, pp. 3–15 (2015)
6. Bäumer, F.S., Grote, N., Kersting, J., Geierhos, M.: Privacy matters: detecting nocuous patient data exposure in online physician reviews. In: Proceedings of the 23rd ICIST, Druskininkai, Lithuania, vol. 756, pp. 77–89 (2017)
7. Beck, A.J.: Nutzung und Bewertung deutscher Arztbewertungsportale durch Patienten in deutschen Hausarztpraxen, Ulm University (2014)

8. Dpa, Niederlage für Bewertungsportal: Ärztin klagt erfolgreich (2018). https://www.abendblatt.de/ratgeber/gesundheit/article213495565/Niederlage-fuer-Bewertungsportal-Aerztin-klagt-erfolgreich.html. Accessed 26 Feb 2018
9. Emmert, M., Meier, F., Heider, A.-K., Dürr, C., Sander, U.: What do patients say about their physicians? an analysis of 3000 narrative comments posted on a German physician rating website. Health Policy **118**(1), 66–73 (2014)
10. Emmert, M., Meier, F., Pisch, F., Sander, U.: Physician choice making and characteristics associated with using physician-rating websites: cross-sectional study. JMIR **15**(8), e187 (2013)
11. Emmert, M., Sander, U., Esslinger, A.S., Maryschok, M., Schöffski, O.: Public reporting in Germany: the content of physician rating websites. Methods Inf. Med. **51**(2), 112–120 (2012)
12. Emmert, M., Sander, U., Pisch, F.: Eight questions about physician-rating websites: a systematic review. JMIR **15**(2), e24 (2013)
13. Erentaitė, R., Bergman, L.R., Žukauskienė, R.: Cross-contextual stability of bullying victimization: a person-oriented analysis of cyber and traditional bullying experiences among adolescents. Scand. J. Psychol. **53**(2), 181–190 (2012)
14. European Commission, Digital single market (2018). https://ec.europa.eu/commission/priorities/digital-single-market_en#background. Accessed 26 Feb 2018
15. Fischer, S., Emmert, M.: A review of scientific evidence for public perspectives on online rating websites of healthcare providers. In: Gurtner, S., Soyez, K. (eds.) Challenges and Opportunities in Health Care Management, pp. 279–290. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-12178-9_22
16. Gao, G.G., McCullough, J.S., Agarwal, R., Jha, A.K.: A Changing landscape of physician quality reporting: analysis of patients' online ratings of their physicians over a 5-year period. JMIR **14**(1), e38 (2012)
17. Geierhos, M., Bäumer, F.S.: Erfahrungsberichte aus zweiter Hand: Erkenntnisse über die Autorschaft von Arztbewertungen in Online-Portalen, Book of Abstracts der DHd-Tagung 2015, Graz, Austria, pp. 69–72 (2015)
18. Geierhos, M., Bäumer, F.S., Schulze, S., Klotz, C.: Understanding the patient 2.0. In: Christiansen, H., Stojanovic, I., Papadopoulos, George A. (eds.) CONTEXT 2015. LNCS (LNAI), vol. 9405, pp. 159–171. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-25591-0_12
19. Geierhos, M., Bäumer, F.S., Schulze, S., Stuß, V.: 'I grade what I get but write what I think.' Inconsistency analysis in patients' reviews. In: Proceedings of the European Conference on Information Systems, Münster, Germany, pp. 1–15 (2015)
20. Hanauer, D.A., Zheng, K., Singer, D.C., Gebremariam, A., Davis, M.M.: Public awareness, perception, and use of online physician rating sites. JAMA **311**(7), 734–735 (2014)
21. Hinz, V., Drevs, F., Wehner, J.: Electronic Word of Mouth about Medical Services, no. 2012/05, HCHE Research Paper (2012)
22. Hu, N., Pavlou, P.A., Zhang, J.: Can online reviews reveal a product's true quality?: empirical findings and analytical modeling of online word-of- mouth communication. In: Proceedings of the 7th ICTD, ACM, Ann Arbor, MI, USA, pp. 324–330 (2016)
23. Horton, J., Golden, J.: Reputation Inflation: Evidence from an Online Labor Market. New York University, Working Paper (2015)
24. Lagu, T., Lindenauer, P.K.: Putting the public back in public reporting of health care quality. JAMA **304**(15), 1711–1712 (2010)
25. Muth, S.: Russian as a commodity: medical tourism and the healthcare industry in post-Soviet Lithuania, IJB, Informa, vol. 20, pp. 404–416 (2015)
26. Sabin, J.E.: Physician-rating websites. Virtual Mentor **15**(11), 932–936 (2013)

27. Skaržauskienė, R., Tamošiūnaitė, A.: Social media and E-health development in Lithuania. In: Proceedings of the 4th ECSM, Brighton, UK, pp. 497–503 (2014)
28. Stuß, V., Geierhos, M.: Identifikation kognitiver Effekte in Online-Bewertungen. Dhd-Tagung 2015, Graz, Austria (2015)
29. Terlutter, R., Bidmon, S., Röttl, J.: Who uses physician-rating websites? differences in sociodemographic variables, psychographic variables, and health status of users and nonusers of physician-rating websites. JMIR **16**(3), e97 (2014)
30. Worldatlas: Lithuania Facts on Largest Cities (2018). https://www.worldatlas.com/webimage/countrys/europe/lithuania/ltfacts.htm. Accessed 22 Jan 2018
31. Zhu, F., Zhang, X.: The influence of online consumer reviews on the demand for experience goods: the case of video games. In: Proceedings of the 27th ICIS, AIS, Milwaukee, WI, USA, Paper 25 (2006)

Design of an Operator-Controller Based Distributed Robotic System

Uwe Jahn(✉), Carsten Wolff, and Peter Schulz

Institute for the Digital Transformation of Application and Living Domains (IDiAL), Fachhochschule Dortmund, Otto-Hahn-Str. 23, 44227 Dortmund, Germany
uwe.jahn@fh-dortmund.de

Abstract. Modern robots often use more than one processing unit to solve the requirements in robotics. Mobile robots are more and more designed in a modular manner to fulfill the possibility to be extended for future tasks. The usage of multiple processing units leads into a distributed system within one single robot. Therefore, the (software) architecture is even more important than in single-computer robots. The presented DAEbot was designed to implement the Operator-Controller Module (OCM) on a mobile robot. This OCM has been used in other technical systems and splits the system hierarchically into controllers and operator(s). The controllers interact directly with all sensors and actuators within the system. For that reason, hard real-time constrains needs to be complied. The operator however processes the information of the controllers, which can be done by model-based principles using state machines. This paper describes the design of the autonomous mobile DAEbot robot focusing on its architecture with three controllers and two operators as well as the internal communication framework. Furthermore, the simulation capabilities of the system behavior and some safety features are shown.

Keywords: Robotics · Mobile robots · Distributed systems
Operator-Controller module · Health monitoring · Model-based development
Modular systems · Simulation

1 Introduction

Robotic systems are complex technical systems whose design and advancement is a challenge for developers of such systems. Mobile robots e.g. utilize extensive functions as well as a high number of actuators and sensors. Autonomously acting robots need performant hardware to perform complex software algorithms.

Past robots often used one computer as the processing unit of the system. With the evolution for small single board computers (SBC) as more energy efficient and more suitable solutions for processing units of mobile robots, more and more SBC are used. Often, multiple SBCs or microcontrollers are combined to a distributed system to obtain more computing power or better dependability. Furthermore, modular mobile robots like the AMiRo [1] have been released. These modular robots can be extended easily and mostly use multiple processing units.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 59–70, 2018.
https://doi.org/10.1007/978-3-319-99972-2_5

The modular approach and the use of multiple processing units lead into a distributed system within a single robot. Thus, the (software) architecture of those distributed systems within one robot becomes much more important.

One architectural approach for technical systems is the Operator-Controller Module (OCM). The OCM was originally developed for mechatronic systems [2, 3]. In the OCM the system is hierarchically structured into controller and operator parts which e.g. differ in terms of real-time requirements or complexity of the used software.

This paper describes the integration of the OCM architecture approach into the DAEbot. The presented DAEbot (Distributed Architecture Evaluation Robot) (see Fig. 1) was designed as a demonstrator and training platform for distributed systems in robotics. Its modular approach makes it easy to adapt or extend the platform with new sensors and actuators to be prepared for different applications or even new processing units to increase computing power.

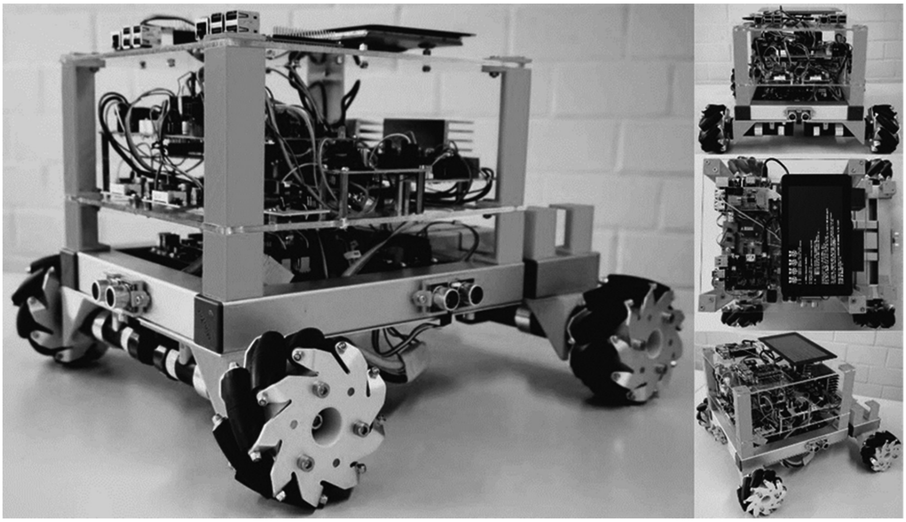


Fig. 1. DAEbot as a demonstrator for an operator-controller based distributed system

This paper is organized as follows: Sect. 2 gives and overview of the concepts and the architecture used in the distributed system. Section 3 describes the design of the DAEbot, including all components and several features. Section 4 shows some results of this project and the last section summarizes the project with a conclusion and outlook to upcoming projects.

2 Concepts and Architecture of the Distributed System

Many design implementations of mobile robots use one single processing unit (e.g. SBC) to control the complex robotic system, one Arduino Uno is used to run a surveillance robot for outdoor security [4], a (not described) microcontroller controls an underwater robot [5] and a mini PC is used to operate a hexapod robot for subsea operations [6].

This architecture of a single processing unit leads into difficulties like combining hard real-time with soft real-time requirements. This architecture also often lacks external surveillance features which can be useful due to safety reasons in mobile robots. Using a single processing unit also prohibits the extension of the system for changing system demands in a modular way.

The DAEbot instead consists of different modules which are setup as a distributed system within one robot. Evaluating a suitable architecture for this distributed system is one of the main goals and the reason for the development of this robot.

In 1979 Ropohl [7] published a three-layer structure for very generalized technical systems (see Fig. 2). The lowest layer (execution system) interacts with the physical world, such as sensors and actuators of a technical system. The information system layer receives information from the execution system. This is used for the information processing of the underlying sensors and actuators. Ropohl's target setting system is another layer which lies hierarchically above the other two layers and is used for planning actions.

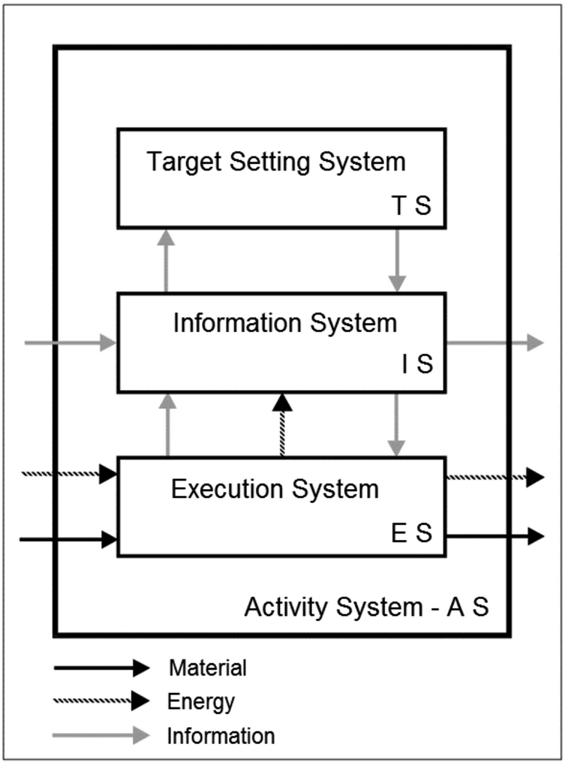


Fig. 2. Ropohl's activity system [7] (Translated from German)

This hierarchically organized structure can also be found in the Operator-Controller Module (OCM).

2.1 Operator-Controller Module

The Operator-Controller Module (OCM) appeared in Naumann's 2000's dissertation as a concept for mixed system- and process based systems for intelligent mechatronic systems [2]. Following Ropohl's approach, the OCM decouples the controller for the direct interaction with the physical system and its sensor- and actuator signals from the operator (see Fig. 3). The operator processes the controllers' information and is able to (re-) configure the controllers. The operator is also used to add non-time-critical features like monitoring to the system.

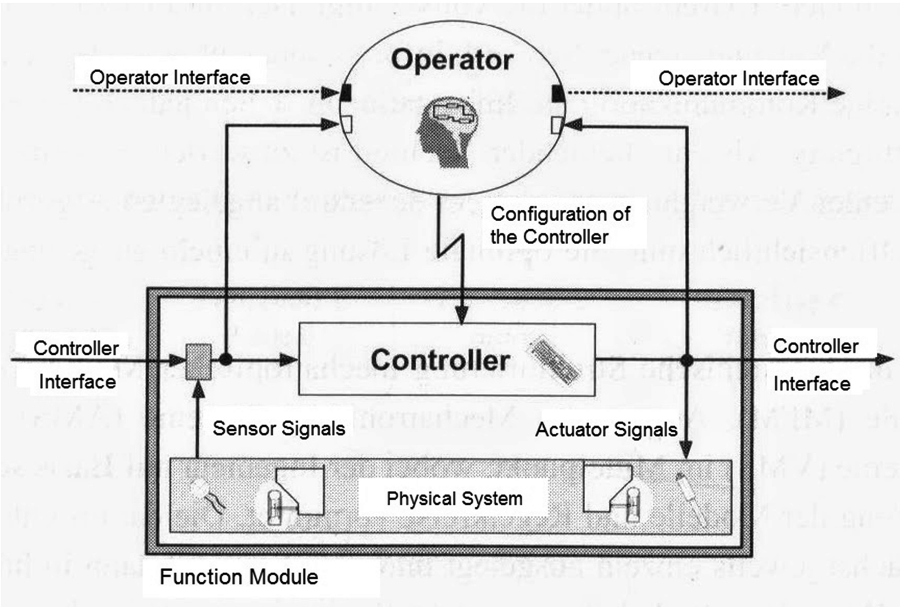


Fig. 3. Concept of the operator-controller module [2] (Translated from German)

2.2 Time-Discrete vs. Event-Driven

A main concept of the Operator-Controller Module is the division into time-discrete and event-driven domains. The controllers are mostly based on time-discrete constraints. The controllers are directly interacting with the systems actuators and sensors and therefore need to guarantee the compliance with hard real-time requirements. In case of an emergency the controller for example needs to stop the systems motors without delays. Furthermore, the controllers need to be self-obtained in case the operator is not available (e.g. fail-safe).

The operator(s) can be implemented without those hard real-time requirements. Information processing depends on the data from the controllers and is therefore using event-driven behavior such as event triggered function calls [3].

This is ideally fitting into IoT and cloud-based systems and opens the path for seamless integration of technical systems into virtual environments – leading to real cyber-physical systems (CPS).

2.3 Model-Based Development

The complexity of autonomous robotic systems, especially when they consist of several components as a distributed system within one robot, is a big challenge for software developers. Therefore, the use of model-based principles is recommended [8, 9].

The operator of the OCM architecture is particularly suitable to use model-based principles as it is mostly triggered by events from the controllers. This event-driven operator can be implemented as statecharts in tools like MATLAB Simulink(Stateflow) [10] or Yakindu SCT [11].

2.4 Software Architectures Based on OCM

The concept of the OCM was originally developed for self-optimizing mechatronic systems [2]. The OCM has been refined later and was used in several projects of the research cluster “Self-optimizing systems in mechanical engineering (SFB614)” [12], e.g. the RailCab project [13]. The OCM approach is also used for the design of metropolitan energy systems [14] or ORC turbines [15].

3 Design of the DAEbot

The DAEbot (Distributed Architecture Evaluation Robot) has been developed to evaluate distributed architectures within one robot, in this case the Operator-Controller Module (OCM). The DAEbot consists of a modular structure, which can be extended with actuators and sensors for different applications, e.g. with a depth camera to navigate autonomously to a given location.

All microcontrollers of the DAEbot are SBCs (single board computers) whose software can be developed independently, without the need to connect them with the robot in the first place. The robot basis is a self-designed platform which is driven by four mecanum wheels [16]. The robot basis contains motor drivers, motor encoders and ultrasonic sensors. All other components added to the robot will be described in this chapter.

3.1 Programming Languages and Methods

As the controllers in OCM needs to work with hard real-time, the controllers should be implemented as efficient as possible. The controllers will also be implemented once and shouldn’t change for future tasks if the connected sensors and actuators won’t change.

Therefore, the development of the controllers is done in C/C++, which complies to compile algorithms efficiently.

In contrast to the controller's source code, the operators' code will change dynamically from one task to another and will be much more complex. For this reason, the development process is assisted by approaches like model-based development. This makes the understanding of the operator's code more intuitive but the generated C/C++ code probably less efficient.

3.2 Communications

The main communication in the DAEbot is done via CAN (Controller Area Network) in publisher subscriber concept. This concept is widely used in robotics e.g. in ROS [17], the de-facto standard framework for robotics. A publisher e.g. a sensor offers its data to the system. A subscriber, e.g. a processing algorithm can subscribe these data for its own purpose.

For this concept, a CAN ID framework has been developed (see Fig. 4). The first bit defines if the attached CAN frame contains sensor data from a controller (C/S = 1) or command data from the operator (C/S = 0). With the next two bits, the priority of the CAN frame can be configured as urgent (00), high (01), medium (10) or low (11). The last eight bits contain a distinct identity for each sensor and actuator value of the system. As a result, a CAN frame can be transmitted to the same entity either to send the sensor data or to switch to a different mode by changing the C/S bit.

Bit	0	1	2	3	4	5	6	7	8	9	10
Subject	C/S	Priority		Identity							
Number of Bits	1	2		8							

Fig. 4. CAN communications ID for the DAEbot

The DAEbot communications framework uses three modes. Mode 0 is used to switch off a publisher, e.g. if the sensor data isn't useful for the operator. Mode 1 (re-) configures a publisher by attributing the sensors sample time. With a mode 2 message, the operator asks the controller for a one-time data transmission.

The publisher subscriber implementation of the DAEbot can be configured to the exact needs of the robot's application in order to use the controller's hardware resources efficiently. Furthermore, the use of the priority bits in the CAN ID helps to prioritize important from less important information, e.g. to prefer real-time controller messages over non-time-critical command messages.

3.3 Modular Design

The DAEbot is based on SBCs which are connected via CAN. This modular approach makes the system future-proof. New SBC's can be added to the system for future tasks e.g. to add more powerful or more efficient SBC's to the system. The modular approach also leads into a modular software structure which can be easily distributed to multiple developers. So, the DAEbot can be programmed in a collaboration and concurrent design process by a number of developers which all focus on their component, bringing everything together as a team when the single tasks are finished.

3.4 Components

The DAEbot utilizes five SBC's as controllers or operators implementing the OCM approach. The three controllers (see Fig. 5, middle layer) are directly connected with several sensors and actuators (bottom layer). The operator and the operator + (top layer) are interacting with the controllers via CAN. To demonstrate the software architecture, the DAEbot itself is designed with layers for the controllers and a layer for the operator (see Fig. 1). As the architecture is modular, more components can easily be added for future tasks.

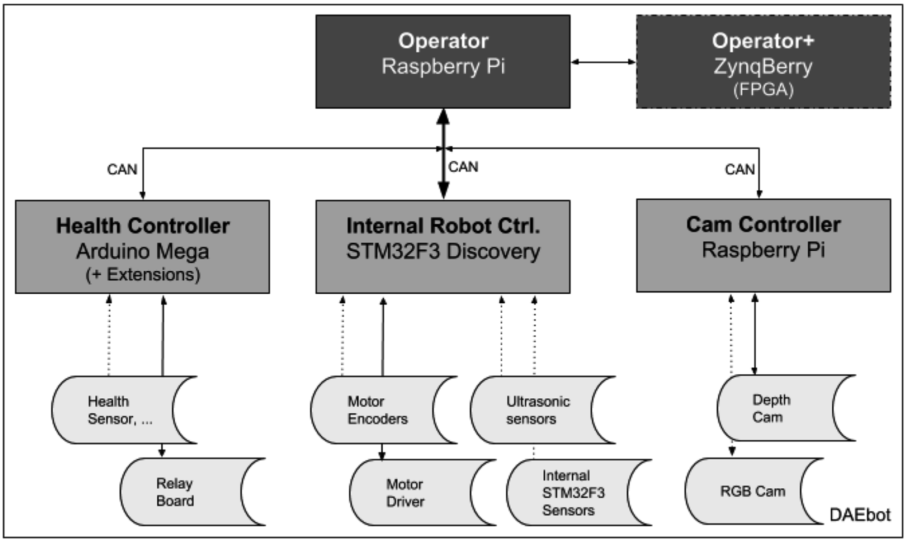


Fig. 5. Components of the DAEbot

Internal Robot Controller

The Internal Robot Controller (IRC) runs the sensors and actuators which are necessary for the movement of the robot. The STM32F3Discovery [18] based controller has been extended with an extension board to add pinouts and a CAN interface to the controller. The IRC receives rotation data from the motor encoders, sensors like accelerometer for odometry and the ultrasonic sensors. The motor drivers are also connected to the STM32F3 board. In order to satisfy the real-time requirements, the microcontroller uses the Operating System FreeRTOS [19]. The IRC implements safety features like to interrupt the motors if the ultrasonic sensors recognize an obstacle movement direction.

Cam Controller

The Raspberry Pi based Cam(era) Controller extends the DAEbot with image sensors, which are used for an autonomous navigation application to demonstrate the robot's capabilities. First, a Raspberry Pi RGB camera is connected to the SBC. Secondly a Microsoft Kinect depth camera is setup. Both cameras use OpenCV to recognize objects or obstacles. A custom linux-based OS for real-time capability has been built with Yocto project [20].

The operator's task is to decide which camera is best for the current situation and task (e.g. in case of darkness the Kinects depth camera would be the obvious choice over the RGB camera). The other camera can be disabled or in case of the Kinect be switched off completely with a relay.

Health Controller

The Health Controller (HC) is used to monitor the condition of the system. It addresses different types of information about the system. This information can be used by the operator(s) to (re-) configure the system behavior, e.g. for critical battery or abnormally high motor temperatures.

First, the HC is collecting information about the energy and battery of the systems. For this, current voltage, current power and the remaining capacity is captured. Second, the temperature of the battery, the motor drivers and the motors are monitored with external temperature sensors. Third, the HC captures air pressure, humidity and temperature of the environment. All these external sensors are set up on a self-developed extension board for an Arduino Mega.

In addition, the HC connects a self-developed relay board with 10 relays which are used to switch off components, like the Kinect or any SBC of the system to shutdown unused hardware for saving energy.

As Arduinos are usually not equipped with an OS, a scheduler was developed to manage the flow of the multiple publishers.

Operator

The main operator of the DAEbot is located on another Raspberry Pi extended with a CAN shield and a touch display to debug and configure the robot. Main task of the Operator is to handle the information from the three underlying controllers. Therefore, state machines are developed with MATLAB Simulink (Stateflow) or Yakindu SCT. These state machines react to incoming CAN frames by performing pre-defined flows e.g. stopping the motor if an obstacle is detected. For this purpose, the system logic was designed model-based in several software modules like battery management or motor logic. All modules combined define the task and behavior of the autonomous robot.

The operator also (re-)configures the controllers' publishers by switching publishers off or set it dynamically to other sample rates due to the current requirements.

In addition, the operator is able to visualize data on the attached display. Especially the data from the Health Controller, like temperatures or environment information can be shown. The touch display is also used to change modes of the robot. It could e.g. be set to an energy-saving mode where the Cam Controller is totally switch off.

Operator+

The goal of the Operator+ is to add hardware resources to the operator whenever its needed, otherwise the Operator+ is switched off with a relay.

The ZynqBerry TE0726-02M SBC has been chosen because of its heterogeneous chip architecture [21].

The ZynqBerry has the form factor of a Raspberry Pi and features a small FPGA SoC and an ARM Cortex-A9 based co-microprocessor. Hence, the Operator+ can take over tasks from the main operator or even runs massively parallel tasks e.g. for computer vision tasks [22] for which an FPGA is more suitable than an ARM based Raspberry Pi.

3.5 Simulation

One benefit of the model-based approach is, that the complex behavior for the autonomous system can be simulated without using the actual robot e.g. by using state machines. The used tools MATLAB Simulink (Stateflow) and Yakindu SCT both offer the ability to simulate the models. In case of the DAEbot the information from the controllers could easily be simulated. So, the operators' behavior can be simulated before deploying it on the hardware. As the software is developed in a modular design, first all features and components can be tested individually. Last, all system dependencies are simulated. This simulation should cover all possible controller inputs to make sure that the system will later work in the real world without problems.

3.6 Safety and Redundancy

A disadvantage of using one single processing unit is the dependency on this unit. If the only processing unit fails, the system itself will fail. Using a distributed system in one robot has the advantage to have other processing units onboard which can bring the system into a save state or even take over tasks of the failing processing unit.

For the DAEbot every component runs a local software watchdog to check if the connection to other components is working fine and can even monitor typical behavior of connected components. If the connection breaks down, the component reacts to this with a predefined strategy. If the Internal Robot Controller e.g. loses the connection to the operators, the controller will shut down all actuators, including the motors. The Health Controller could also be used for self-healing behavior. The controller is able to restart the operator if a dysfunction of it is detected. At restart, the Operator runs self-checking algorithms to solve its dysfunction, e.g. a CAN connection disruption.

In addition, the Health Controller can be used in case of critical conditions like low battery or high temperature. These warnings are processed by the operator(s) and are also visualized on the robots' display.

4 Results

The Operator Controller Module (OCM) promises to fit into the requirements of complex mobile robots like the described DAEbot. In contrast to single processing unit architectures the modular and distributed OCM approach helps to decouple its tasks and features on different levels. The DAEbot uses hard real-time constraints within the described controllers and adds non-time-critical visualization on the operator layer. It also combines embedded C/C++ programming for the controllers with model-based development for the operator layer. This makes the implementation of behavior for such a complex distributed system significantly easier.

The implemented communication framework fits into the architectures main design characteristics with its ability to change the controller's publishing rate by the operator.

Safety and redundancy features can easily be added in the distributed OCM system. With its Health Controller the DAEbot has the ability to monitor the system and environment conditions. This condition information can be used to adapt the distributed system to the robot's tasks in future projects. The Operator+, equipped with an FPGA, has been integrated to run massively parallel algorithms, like computer vision tasks.

5 Conclusion and Outlook

The paper presents the design of a distributed robotic system implementing the Operator-Controller Module (OCM) architecture. The paper outlines the difficulties designing such project, like communication and real-time requirements.

The DAEbot also shows the modular approach for mobile robotics and the use of widely used single board computers like Arduino Mega, STM32 boards or the Raspberry Pi in a distributed system within one robot. The decentralized approach in robots is useful e.g. to design scalable robot platforms which can easily be extended with other components. The OCM architecture is helpful in robotics e.g. to decouple time-critical functions like motor-behavior from non-time-critical functions like monitoring. The paper describes the simulation capabilities of the system behavior and some safety features for the DAEbot.

In a future project, the DAEbot will be refined. The robot e.g. will be equipped with a cognitive operator which is an extended approach of the OCM architecture [23]. This cognitive operator will be designed to work as a cloud service and is used for self-optimizing and planning algorithms. Cloud computing principles, especially for robotics [24] e.g. neural network applications [25] will be evaluated with the DAEbot. The DAEbot will also be used in a multi-robot network [26], e.g. alongside the AMiRo [1].

References

1. Herbrechtsmeier, S., Korthals, T., Schopping, T., Rückert, U.: AMiRo: a modular and customizable open-source mini robot platform. In: 2016 20th International Conference on System Theory, Control and Computing (ICSTCC), Sinaia, pp. 687–692 (2016)

2. Naumann, R.: Modellierung und Verarbeitung vernetzter intelligenter mechatronischer Systeme. Dissertation, Universität Paderborn, 2000, VDI-Verlag, Düsseldorf (Fortschritt-Berichte Reihe 20, Nr. 318) (2000)
3. Oberschelp, O., Hestermeyer, T., Kleinjohann, B., Kleinjohann, L.: Design of self-optimizing agent-based controllers. In: CfPWorkshop 2002 - Agent-Based Simulation 3, Passau, Germany, 7–9 April 2002 (2002)
4. Meghana, S., Nikhil, T.V., Murali, R., Sanjana, S., Vidhya, R., Mohammed, K.J.: Design and implementation of surveillance robot for outdoor security. In: RTEICT 2017 - 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, Proceedings, January 2018, pp. 1679–1682 (2018)
5. Tang, Z., Feng, H.: Design of an ocean current roaming underwater robot. Paper presented at the 2017 4th International Conference on Systems and Informatics, ICSAI 2017, January 2018, pp. 229–233 (2018)
6. Davliakos, I., Roditis, I., Lika, K., Breki, C.-M., Papadopoulos, E.: Design, development, and control of a tough electrohydraulic hexapod robot for subsea operations. Adv. Robot. **32** (9), 477–499 (2018)
7. Ropohl, G.: Allgemeine Technologie, Eine Systemtheorie der Technik, 3. überarbeitete Auflage, Universitätsverlag Karlsruhe, Germany (2009)
8. Forschungsunion Wirtschaft – Wissenschaft und acatech – Deutsche Akademie der Technikwissenschaften e.V., "Umsetzungsempfehlungen für das Zukunftprojekt Industrie 4.0", pp. 28–30 (2013)
9. Lauschner, U., Igel, B., Krawczyk, L., Wolff, C.: Applying model-based principles on a distributed robotic system application. In: 2015 IEEE 8th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Warsaw, pp. 893–897 (2015)
10. MathWorks Homepage. https://www.mathworks.com/products/simulink.html. Accessed 20 Mar 2018
11. Yakindu SCT Homepage. https://www.itemis.com/en/yakindu/state-machine/. Accessed 20 Mar 2018
12. Adelt, P., et al.: Selbstoptimierende Systeme des Maschinenbaus – Definitionen, Anwendungen, Konzepte. HNI- Verlagsschriftenreihe, Band 234, Paderborn, Germany (2008)
13. Gausemeier, J., Steffen, D., Donoth, J., Kahl, S.: Conceptual design of modularized advanced mechatronic systems. In: 17th International Conference on Engineering Design (ICED 2009), 24–27 August 2009, Stanford, CA, USA (2009)
14. Wolff, C., Lippmann, T., Lauschner, U.: Systems engineering for metropolitan energy systems – Ruhrvalley. In: 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, Bukarest, Romania (2017)
15. Wolff, C., et al.: Software architecture for an ORC turbine – case study for an intelligent technical system in the era of the Internet of Things. In: Damaševičius, R., Mikašytė, V. (eds.) ICIST 2017. CCIS, vol. 756, pp. 226–237. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67642-5_19
16. Rohrig, C., Hes, D., Kunemund, F.: Motion controller design for a mecanum wheeled mobile manipulator. In: 2017 IEEE Conference on Control Technology and Applications (CCTA), pp. 444–449 (2017)
17. ROS Homepage. http://wiki.ros.org/ROS/Concepts. Accessed 26 Mar 2018
18. ST Homepage. http://www.st.com/en/evaluation-tools/stm32f3discovery.html. Accessed 26 Mar 2018
19. Barry, R.: Mastering the FreeRTOS Real Time Kernel, 1st edn., L. R. T. Engineers, Ed (2016)

20. Yocto Project Homepage. https://www.yoctoproject.org/. Accessed 21 Mar 2018
21. Trenz Electronic Homepage. https://wiki.trenz-electronic.de/display/PD/TE0726+-+ZynqBerry. Accessed 22 Mar 2018
22. Irwansyah, A., et al.: FPGA-based generic architecture for rapid prototyping of video hardware accelerators using NoC AXI4-stream interconnect and GigE vision camera interfaces. Presented at the Bildverarbeitung in der Automation (BVAu) 2014, Lemgo, Germany (2014)
23. Oberschelp, O., Hestermeyer, T., Giese, H.: Strukturierte Informationsverarbeitung für selbstoptimierende mechatronische Systeme. In: Gausemeier, J., Wallaschek, J. (Hrsg.) 2. Paderborner Workshop Intelligente Mechatronische Systeme, 25–26 März 2004, HNI-Verlagsschriftenreihe, Band 145, Paderborn (2004)
24. Wan, J., Tang, S., Yan, H., Li, D., Wang, S., Vasilakos, A.V.: Cloud robotics: current status and open issues. IEEE Access **4**, 2797–2807 (2016)
25. Da Silva, I.J., Vilao, C.O., Costa, A.H.R., Bianchi, R.A.C.: Towards robotic cognition using deep neural network applied in a goalkeeper robot. In: 2017 Latin American Robotics Symposium (LARS) and 2017 Brazilian Symposium on Robotics (SBR), Curitiba, pp. 1–6 (2017)
26. Iocchi, L., Nardi, D., Salerno, M.: Reactivity and Deliberation: A Survey on Multi-Robot Systems. Balancing Reactivity and Social Deliberation in Multi-Agent Systems, From RoboCup to Real-World Applications (Selected Papers from the ECAI 2000 Workshop and Additional Contributions), pp. 9–34 (2001)

Information Systems: Special Session on Information and Software Technologies for Intelligent Power Systems

Deployment of Battery Swapping Stations for Unmanned Aerial Vehicles Subject to Cyclic Production Flow Constraints

G. Bocewicz[1(✉)], P. Nielsen[2], Z. Banaszak[1], and A. Thibbotuwawa[2]

[1] Faculty of Electronics and Computer Science, Koszalin University of Technology, Koszalin, Poland
bocewicz@ie.tu.koszalin.pl

[2] Department of Materials and Production, Aalborg University, Aalborg, Denmark
{peter,amila}@mp.aau.dk

Abstract. Given is a production system in which material handling operations are carried out by a fleet of UAVs. A problem has been formulated for this case of cyclic multi-product batch production flow, which combines the problems of split delivery-vehicle routing with time windows and deployment of battery swapping depots. It is assumed that the times of execution of pickup and delivery operations are known. During these operations, workpieces following different production routes reach and leave workstations cyclically. Given is the number of battery swapping depots and their potential arrangement. Given is also the rate of power consumption by an UAV in hovering mode or flying at a constant speed as well as during take-off and landing. The goal is to find the number of UAVs and the routes they fly to serve all the workstations periodically, within a given takt time, without violating constraints imposed by the due-time pickup/delivery operations and collision-free movement of UAVs. A declarative model of the analysed case allows to view the problem under consideration as a constraint satisfaction problem and solve it in the Oz Mozart programming environment.

Keywords: Unmanned aerial vehicles · Battery swapping · Routing problem

1 Introduction

Unmanned aerial vehicles (UAVs) have the potential to significantly reduce the cost and time required to deliver commodities. Delivering with UAVs may be faster than delivering with traditional delivery vehicles, as drones are not limited by established infrastructure such as roads, and generally face less complex obstacle avoidance scenarios [11, 17]. UAVs are more and more frequently considered for use in the movement of materials and products between and within departments and even between workstations [13]. One of the most important factors limiting the use of UAVs in this type of applications is the weight and limited capacity of their batteries. To eliminate the limitation imposed by the finite energy source on board of each UAV, a network of shared refuelling stations distributed across the field is frequently proposed.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 73–87, 2018.
https://doi.org/10.1007/978-3-319-99972-2_6

To accomplish a mission, an UAV can refuel at any station and return to service. The refuelling means charging or swapping the batteries onboard an UAV.

Many of those solutions involve downtime, as an UAV has to dock at charging stations or when "cold" battery swapping is used (techniques which require a complete shutdown of the vehicle's onboard electronics as the spent battery is being swapped for a new one). However, there are also novel solutions which enable a "hot" battery swap, allowing the vehicle to remain powered on throughout the battery changing process. In this paper, we discuss solutions based on fast hot-swaps of batteries, which ensure efficient execution of missions.

In this context, further considerations focus on routing methods for locating a minimum number of "hot" battery swap depots for a fleet of UAVs cyclically transporting workpieces among a set of workstations. To put it differently, this study explores the problem of deployment (arrangement) of battery swapping depots as part of a split delivery vehicle routing problem with time windows, which takes into account the constraints imposed by the cyclic nature of the production flow system in which the fleet is used. The focus of the present study are solutions which minimize the number of swapping depots to be used within a given production takt time. Since UAV routing and scheduling problems are known to be NP-hard, the literature related to these problems primarily investigates various heuristic methods or metaheuristic-based search strategies. This fact justifies the adoption of a declarative modelling framework in this study as well as the use of a constraint programming environment (OzMozart) to solve the above problem. The results fall within the scope of research, reported in previous papers, on energy consumption driven models for UAV routing [1, 17].

The remainder of this paper is organized as follows: the key issues concerning the delivery of workpieces by battery-powered UAVs are described in Sect. 2, which is followed by a case study of deployment of battery swapping stations in Sect. 3. Section 4 presents a declarative model of the discussed UAV fleet routing problem focused on minimalizing the number of battery swapping stations. The model allows to consider the problem of UAV routing as a constraint satisfaction problem and, consequently, to solve it in the Oz Mozart environment. The key conclusions are formulated and the main directions of future research are suggested in Sect. 5.

2 Related Work

The new technologies enabling the construction of strong, lightweight airframes and the invention of lithium polymer batteries, with their relatively high energy density, have significantly expanded the range of applications of new generation UAVs [2, 4, 6, 10, 15]. In service industry, the novel possibilities offered by technology call for new solutions regarding product distribution routing, maintenance operations routing, e-delivery operations routing, and so on. The research problems considered are limited by the specific nature of the needs that these problems address. All of them, however, grow out of the common schema of the Vehicle Routing Problem (VRP) [5, 7, 14, 18] which is a generalization of the well-known travelling salesman problem. In a VRP for a given fleet of UAVs and a set of customers, the goal is to find a set of routes with a minimum total length which guarantee timely delivery. Determining the optimal solution to a VRP is NP-hard.

In general, VRP-driven approaches used for routing of UAVs can be grouped into following categories [17]:

- periodicity (the cyclic manner in which deliveries are performed),
- multiple depots (different depots supplying different customers),
- time windows,
- pickup and delivery,
- capacity (limited carrying capacities of UAVs),
- refuelling stations,
- and others.

Various combinations of the above-mentioned aspects are reflected in the formulations of specific UAV routing problems encountered in practice. The problems that are most frequently described in the literature are exploration, surveillance, reconnaissance, and delivery problems. The problem of area search or exploration has been studied in various contexts with different objectives [3]. Its main objective is to explore an unknown area within the duration of a fixed mission of a given fleet of UAVs. The problem is to define cooperative trajectories of the vehicles, such that the cumulative, area explored is maximized subject to a constraint on mission duration, which is given and fixed in advance.

The main aim of the surveillance and reconnaissance routing problems is to find routes which will allow vehicles to collect data from a set of task locations and deliver them to a control station. In those problems, the objective is to minimize the maximum time of delivery of all task data to the control station, while satisfying the revisit period constraints of each task [9].

Moreover, in cases in which a region is expected to have "pop-up" targets, the formulation of the problem has to be extended to encompass searching for the targets. This type of problem, which is called a persistent surveillance problem, differs from the exploration problem, as it involves continuous/repeated coverage of the target space, minimizing the time between re-visits. Its goal is to find sequences of visits to discrete sites in a periodic fashion [12]. The necessity of periodically monitoring selected targets, which is associated with the concept of time window constraints, places the problem in the class of Vehicle Routing Problems with Time Windows, which are encountered very frequently in settings where decisions about the distribution of goods and services have to be made. The problem involves a fleet of vehicles which set off from a depot to serve a number of customers at different geographic locations who have various demands. The missions have to be completed within specific time windows before the vehicles return to the depot. The objective of the problem is to find routes for the vehicles to serve all the customers at a minimum cost (in terms of travel, distance, etc.) without violating the capacity and travel time constraints of the vehicles and the time window constraints set by the customers [16].

The constraint satisfaction problem considered in this present study can be seen as a kind of VRP whose goal is to find the most cost-effective routing of a fleet of vehicles which periodically visit a number of workstations, picking up/delivering workpieces within a given time period, subject to constraints imposed by multiple battery swap depots, multi-commodity flow, capacitated pickup/delivery, energy consumption, and split delivery. In focusing on the choice of routes that will minimize production takt time and the number of battery swapping stations, the problem ignores the factors key to UAV delivery such as vehicle capacity, battery weight, changing payload weight, and reuse of vehicles to reduce costs, as well as kinematics and dynamics constraints [8]. It is worth adding that the majority of studies which take account of kinematics and dynamics constraints of UAVs are aimed at developing models allowing one to calculate the total power consumption of UAVs in different flight scenarios including horizontal/vertical motion and hovering.

The case considered above can be extended to problems whose goal is to minimize cost or delivery time while taking into account battery weight, payload weight, and drone reuse, e.g. problems which seek to optimize the number of drones, the routes they fly, as well as their battery weight, payload weight, and energy consumption. The same is true of the problem of deployment of battery swapping/charging depots. Assuming that the battery swapping stations are mobile, i.e. that batteries can be swapped at stations located on vehicles of a ground fleet, a new cooperative routing problem is proposed for a fleet of battery swapping vehicles travelling on the ground and a fleet of UAVs moving in the air above them.

3 Deployment of Battery Swapping Stations

3.1 Illustrative Example

In a flow production system consisting of five workstations, whose structure is shown in Fig. 1, two different products are manufactured at the same time. The technological route of product J_1, marked in red, runs through workstations R_2, R_1, and R_4, with respective technological operation times of 20 u.t. (units of time) for $O_{1,1}$, 20 u.t. for $O_{1,2}$ and 25 u.t. for $O_{1,3}$. In turn, the technological route of product J_2, marked in blue, runs through workstations R_1, R_3, and R_5 whose respective technological operation times are 15 u.t. for $O_{2,1}$, 30 u.t. for $O_{2,2}$ and 20 u.t. for $O_{2,3}$. When travel times

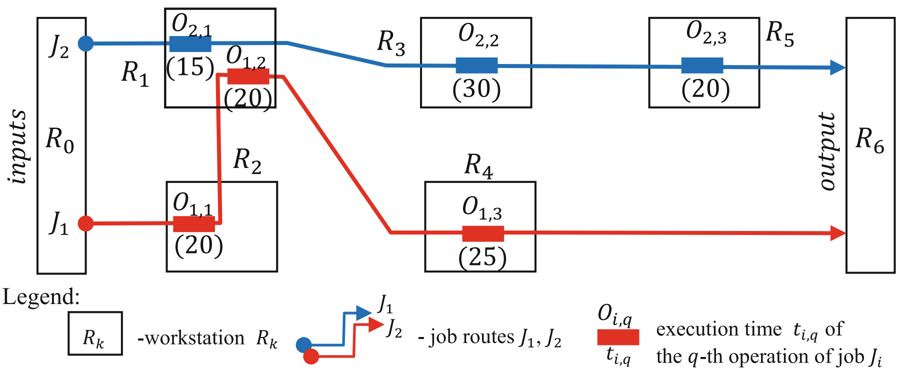


Fig. 1. Structure of a multi-item batch flow production system [source: personal collection]

between workstations, exchange-of-die times, and loading/unloading times are omitted, it is easy to notice that workstation R_1, which is the bottleneck in the production flow, determines production takt time $TP = 35$, see Fig. 2. In this study, production takt time TP is understood as the time that elapses between two successive items of a given product coming out from the production process.

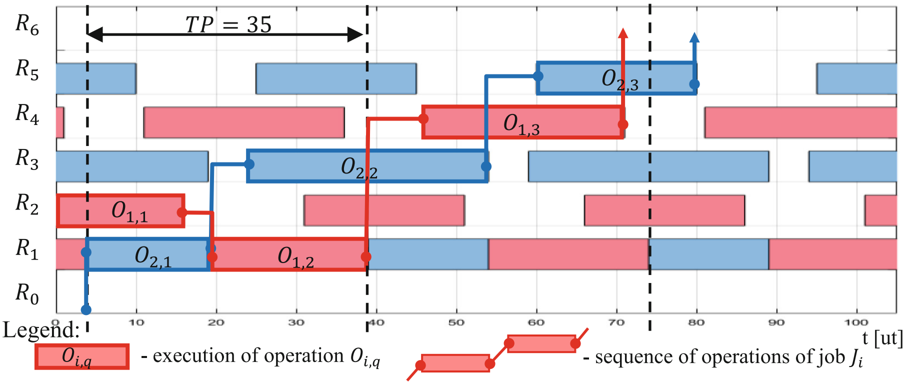


Fig. 2. Gantt's chart of production flow for zero values of transport time between workstations, changeover time, and loading/unloading time [source: personal collection]

Let us assume that transport operations $o_1, \ldots, o_8$ are carried out by a fleet of four UAVs U_1, U_2, U_3, and U_4. The trajectories of the cyclic flights of drones U_1 and U_2 (drones U_3 and U_4 move along the same trajectory as U_2) are shown in Fig. 3. Let us also assume that the time of each operation o_α and the times of service flights (between successive operations o_k, o_l along the trajectory of a cyclically repeated mission) are set depending on the requirements of the production process. Examples of courses of trajectories of the UAV fleet are shown in Fig. 3, and the corresponding pickup/delivery operations are presented in the Gantt chart of production flow in Fig. 4. The Gantt chart presents an optimal solution (guaranteeing the shortest production takt-time) for a fleet of four drones. Compared to the chart in Fig. 2, the production takt time is longer by 3 u.t. ($TP = 38$ u.t.), see Fig. 4.

The fact that it is impossible to obtain a production takt time of $TP = 35$ (Fig. 2) in this system results from the assumptions of mutual exclusion (only one drone can land at/take off from a workstation at any time) and a lack of additional service stations (an UAV either visits one of the resources or travels between them).

In practice, however, such a solution is not always feasible. There are a number of production- and transport-related factors which condition the possibility of executing such a schedule. Worth mentioning here is the need to take into consideration potential collisions between UAVs which use the same corridors (cases considered in [1]), and the need for access to an energy source guaranteeing continued operation of UAVs.

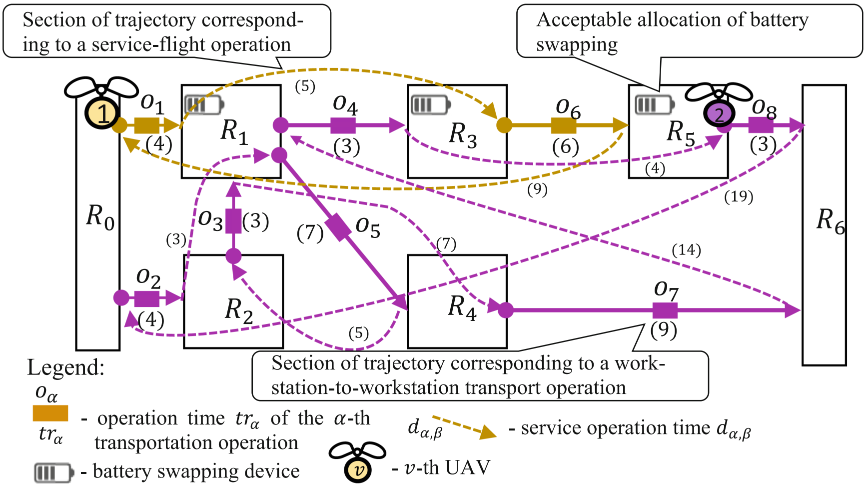


Fig. 3. Schematic layout of considered workshop [source: personal collection]

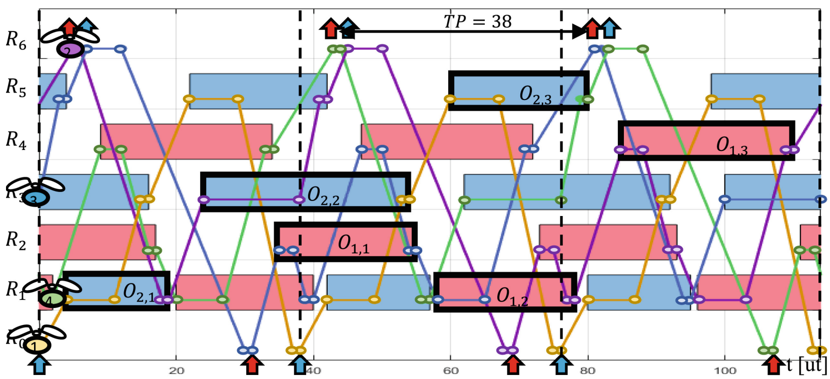


Fig. 4. Gantt chart of production flow taking into account material handling operations, service flight operations, and hovering of UAVs [source: personal collection]

3.2 Problem Description

The example discussed in Sect. 3.1 demonstrates that the value of production takt time depends on the times of technological operations and the times related to the movement and placement of products, as well as workstation changeovers. This means that if the originally adopted takt time value of $TP = 36$ u.t. is to remain the same, given that material handling operation times have non zero values, alternative ways of organizing the handling operations performed by UAVs should be looked for.

In this context, the problem considered assumes a given set of workstations allocated at different points, to be served periodically (in a specified time window) by several capacitated UAVs charged from a set of spatially distributed battery charging depots. The goal is to minimize the number of UAVs needed, such that each customer is reached with the right delivery at the right moment of time and the capacity and battery change or charge constraints are satisfied. A secondary objective is to minimize the total distance travelled. An UAV routing and scheduling problem like this is known to be NP-hard.

4 Declarative Modelling

4.1 Problem Statement

A mathematical formulation of the model considered:

Symbols:

R_k: resource k;
$O_{i,q}$: operation q of J_i;
o_α : transport operation α;
pb_v: battery capacity of transport means v (v-th UAV);
b_α: index of transport operation which precedes o_α;
f_α: index of transport operation which follows o_α
J_i: job i;
U_v: transport means v (v-th UAV);
$K_{\mu,\varepsilon}$: corridor linking resources R_μ and R_ε;

Sets and sequences:

R: the set of resources R_k (workstations);
K: the set of links $K_{\mu,\varepsilon}$ (corridors between workstations);
J: the set of jobs J_i, (production processes);
O_i: sequence of operations for J_i: $O_i = \left(O_{i,1}, \ldots, O_{i,q}, \ldots, O_{i,lm_i}\right)$;
p_i: route of J_i, sequence of resources on which operations $O_{i,q}$ are executed: $p_i = \left(p_{i,1}, \ldots, p_{i,q}, \ldots, p_{i,lm_i}\right)$, $p_{i,q} \in R$;
Q_k: the set of operations executed on R_k;
$\mathcal{O}$: the set of transport operations o_α;
S_k: the set of transport operations started from R_k, $S_k \subseteq \mathcal{O}$;
E_k: the set of transport operations ending on R_k, $E_k \subseteq \mathcal{O}$;
U: the set of transport means U_v (transport processes);
B: sequence of predecessor indices of transport operations, $B = (b_1, \ldots, b_\alpha, \ldots, b_\omega)$, $b_\alpha \in \{0, \ldots \omega\}$;
F: sequence of successor indices of transport operations, $F = (f_1, \ldots, f_\alpha, \ldots, f_\omega)$, $f_\alpha \in \{1, \ldots \omega\}$.
PB: sequence of UAV battery capacity values: $PB = (pb_1, \ldots, pb_v, \ldots, pb_l)$, pb_v - battery capacity of v-th UAV

Parameters:

m: number of resources;
n: number of jobs;
lm_i: number of operations of J_i;
$t_{i,q}$: operation time of $O_{i,q}$;
$d_{a,\beta}$: travel time between resource at which operation o_α ends and resource at which operation o_β begins;
zp: battery consumption (per unit of time) by an UAV waiting on resource (hovering of an UAV);
zs: battery consumption (per unit of time) during a service flight;
zl: battery consumption (per unit of time) during a transport operation;
tw: battery replacement time (the same for all UAVs)
TP^*: maximum value of production takt time TP
h: number of links;
l: number of transport means;
ω: number of transport operations;
tr_α: operation time of o_α,

Variables:

TP: production takt time;
$y_{i,q}$: end time of operation $O_{i,q}$;
yt_α: end time of operation o_α;
xs_α: the moment the resource occupied by an UAV is released after completion of operation o_α;
lb_α: charge level of UAV battery after the completion of operation o_α;
$\mathcal{O}_\mathcal{B}$: a subset of transport operations followed by battery replacement: $\mathcal{O}_\mathcal{B} \subseteq \mathcal{O}$;
b_α: index of the transport operation preceding operation o_α (operations o_{b_α} and o_α are executed by the same UAV); $b_\alpha = 0$ means that o_α is the first operation of the system cycle;
f_α: index of the transport operation following o_α, (operations o_α and $o_{f\alpha}$ are executed by the same UAV)
$x_{i,q}$: start time of operation $O_{i,q}$,
xt_α: start time of operation o_α;

Constraints:

I. For job operations (production processes):

$$y_{i,q} = x_{i,q} + t_{i,q}, \quad q = 1\ldots lm_i, \quad \forall J_i \in J, \tag{1}$$

$$y_{i,q} \leq x_{i,q+1}, \quad q = 1\ldots(lm_i - 1), \quad \forall J_i \in J, \tag{2}$$

$$y_{i,q} \leq x_{i,q} + TP, \quad q = 1\ldots lm_i, \quad \forall J_i \in J, \tag{3}$$

$$\left(y_{i,a} \leq x_{j,b}\right) \vee \left(y_{j,b} \leq x_{i,a}\right), \text{ when } O_{i,a}, O_{j,b} \in Q_k, \forall R_k \in R, \tag{4}$$

$$TP \leq TP^*. \tag{5}$$

II. For UAVs (transport process operations):

$$yt_\alpha = xt_\alpha + tr_\alpha, \quad \alpha = 1, 2, \ldots, \omega, \tag{6}$$

$$b_\alpha = 0, \quad \forall\alpha \in BS, BS \subseteq BI = \{1, 2, \ldots, \omega\}, |BS| = l \tag{7}$$

$$b_\alpha \neq b_\beta \quad \forall\alpha, \beta \in BI \backslash BS, \quad \alpha \neq \beta, \tag{8}$$

$$f_\alpha \neq f_\beta \quad \forall\alpha, \beta \in BI, \ \alpha \neq \beta, \tag{9}$$

$$(b_\alpha = \beta) \Rightarrow \left(f_\beta = \alpha\right), \forall b_\alpha \neq 0, \tag{10}$$

$$\left[(b_\alpha = \beta) \wedge \left(b_\beta \neq 0\right)\right] \Rightarrow \left(yt_\beta + d_{\beta,\alpha} \leq xt_\alpha\right), \alpha, \beta = 1, 2, \ldots, \omega, \tag{11}$$

$$\left[(f_\alpha = \beta) \wedge \left(b_\beta = 0\right)\right] \Rightarrow \left(yt_\alpha + d_{\alpha,\beta} \leq xt_\beta + TP\right), \alpha, \beta = 1, 2, \ldots, \omega, \tag{12}$$

$$xs_\alpha \geq yt_\alpha, \quad \alpha = 1, 2, \ldots, \omega, \tag{13}$$

$$\left[(f_\alpha = \beta) \wedge \left(b_\beta \neq 0\right)\right] \Rightarrow \left(xs_\alpha = xt_\beta - d_{\alpha,\beta}\right), \ \alpha, \beta = 1, 2, \ldots, \omega, \tag{14}$$

$$\left[(f_\alpha = \beta) \wedge \left(b_\beta = 0\right)\right] \Rightarrow \left(xs_\alpha = xt_\beta - d_{\alpha,\beta} + TP\right), \ \alpha, \beta = 1, 2, \ldots, \omega, \tag{15}$$

$$\begin{gathered}\left[\left(xs_\alpha < yt_\beta\right) \wedge (xs_\beta - TP < yt_\alpha)\right] \vee \left[\left(xs_\beta < yt_\alpha\right) \wedge (xs_\alpha - TP < yt_\beta)\right], \\ \forall o_\alpha, o_\beta \in S_k, \ k = 1, \ldots, m,\end{gathered} \tag{16}$$

$$\begin{gathered}\left[\left(xs_\alpha < yt_\beta\right) \wedge (xs_\beta - TP < yt_\alpha)\right] \vee \left[\left(xs_\beta < yt_\alpha\right) \wedge (xs_\alpha - TP < yt_\beta)\right], \\ \forall o_\alpha, o_\beta \in E_k, \ k = 1, \ldots, m,\end{gathered} \tag{17}$$

$$\begin{gathered}\left[\left(xs_\alpha < xt_\beta\right) \wedge (xt_\beta - TP < yt_\alpha)\right] \vee \left[\left(xt_\beta < yt_\alpha\right) \wedge (xs_\alpha - TP < yt_\beta)\right], \\ \forall o_\alpha \in E_k, \ \forall o_\beta \in S_k \, k = 1, \ldots, m.\end{gathered} \tag{18}$$

III. For UAV batteries:

$$lb_\alpha = pb_v \, \forall o_\alpha \in \mathcal{O}_\mathcal{B}, \ U_v \text{ executes of } o_\alpha \tag{19}$$

$$\begin{gathered}lb_\alpha = lb_\beta - zp\left(xs_\beta - yt_\beta\right) - zs\left(xt_\alpha - xs_\beta\right) - zl(yt_\alpha - xt_\alpha), \\ \forall o_\alpha \notin \mathcal{O}_\mathcal{B}, f_\beta = \alpha,\end{gathered} \tag{20}$$

$$lb_\alpha > 0,\ \forall o_\alpha \in \mathcal{O}, \tag{21}$$

IV. For transport and production processes (linking UAVs with jobs)

$$x_{i,q} = yt_\alpha + c \times TP,\ c \in \mathbb{N},\ \forall o_\alpha \in E_k,\ \forall O_{i,q} \in Q_k,\ k = 1, \ldots, m, \tag{22}$$

$$y_{i,q} = xt_\alpha + c \times TP,\ c \in \mathbb{N},\ \forall o_\alpha \in S_k,\ \forall O_{i,q} \in Q_k,\ k = 1, \ldots, m. \tag{23}$$

For a given UAV fleet (set U), the following question is considered: *Does there exist a set of routes (represented by sequences B, F) and a set of battery swapping operations (set $\mathcal{O}_\mathcal{B}$) as well as a deployment of corresponding battery swapping devices that guarantees the existence of a production schedule $(x_{i,q}, xt_\alpha)$ satisfying constraints imposed by the given takt time?*

The above problem can be formulated as a Constraint Satisfaction Problem:

$$CS = (\mathcal{V}, \mathcal{D}, \mathcal{C}) \tag{24}$$

where: $\mathcal{V} = \{B, F, \mathcal{O}_\mathcal{B}, X, XT, LB\}$ is a set of decision variables, where $X = \{x_{i,q} | i = 1 \ldots n, q = 1 \ldots lm_i\}$, $XT = \{xt_\alpha | \alpha = 1, 2, \ldots, \omega\}$, $LB = \{lb_\alpha | \alpha = 1, 2, \ldots, \omega\}$; $\mathcal{D}$ is a discrete finite set of domains of variables $\mathcal{V}$; $\mathcal{C}$ is a set of constraints describing the following relations: the execution order of job operations (1)–(3) and UAV operations (22), (23); exclusion of job operations (4) and UAV operations performed on shared resources (16)–(18). These constraints ensure cyclic routes (7)–(10), nonempty batteries of UAVs (19)–(21), and determine the execution order of transport operations (6), (11)–(15) and production takt time requests (5).

To solve the problem formulated as CS (24), one must determine such values (determined by $\mathcal{D}$) of decision variables B, F (UAV routes), $\mathcal{O}_\mathcal{B}$ (set of swapping operations) X, XT (production schedules and transport operation schedules), and LB (charge levels of UAV batteries), for which all the constraints $\mathcal{C}$ (including the mutual exclusion constraint, the cyclic operation execution constraint, etc.) will be satisfied.

In the special case where routes (B, F) or operations followed by battery replacement $(\mathcal{O}_\mathcal{B})$ are known, the above problem can be reduced to the following two subproblems:

Subproblem 1: For a given set of commodity delivery routes (B, F), the following question is considered: Does there exist a set of battery swapping operations (set $\mathcal{O}_\mathcal{B}$) guaranteeing the existence of a production schedule $(x_{i,q}, xt_\alpha)$ that satisfies the constraints imposed by the given takt time?

$$CS_R = (\mathcal{V}_R, \mathcal{D}_R, \mathcal{C}_R) \tag{25}$$

where: $\mathcal{V}_R = \{\mathcal{O}_\mathcal{B}, X, XT, LB\}$ is a set of decision variables, $\mathcal{D}_R$ is a discrete finite set of domains of variables $\mathcal{V}_R$; $\mathcal{C}_R = \mathcal{C} \cup \mathcal{C}_{FB}$, and $\mathcal{C}_{FB}$ is a set of constraints describing given UAV routes (determined by B, F).

Subproblem 2: For a given set of battery swapping operations (set $\mathcal{O}_{\mathcal{B}}$), the following question is considered: Does there exist a set of commodity delivery routes (B, F) guaranteeing the existence of a production schedule $(x_{i,q},\ xt_{\alpha})$ that satisfies the constraints imposed by the given takt time?

$$CS_{CH} = (\mathcal{V}_{CH}, \mathcal{D}_{CH}, \mathcal{C}_{CH}) \tag{26}$$

where: $\mathcal{V}_{CH} = \{B, F, X, XT, LB\}$ is a set of decision variables, $\mathcal{D}_{CH}$ is a discrete finite set of domains of variables $\mathcal{V}_{CH}$; $\mathcal{C}_{CH} = \mathcal{C} \cup \mathcal{C}_{\mathcal{O}}$, and $\mathcal{C}_{FB}$ is a set of constraints describing the charging process based on operation $\mathcal{O}_{\mathcal{B}}$.

These types of problems are typically solved using constraint programming CP/CLP environments, such as Oz Mozart, IBM ILOG, or ECL$^{\mathrm{i}}$PS$^{\mathrm{E}}$.

4.2 Computational Results

The solutions sought to the organization of transport between workstations regarding the set of commodity delivery routes and the quantity of battery swapping devices and their spatial deployment, should guarantee execution of production flow within the

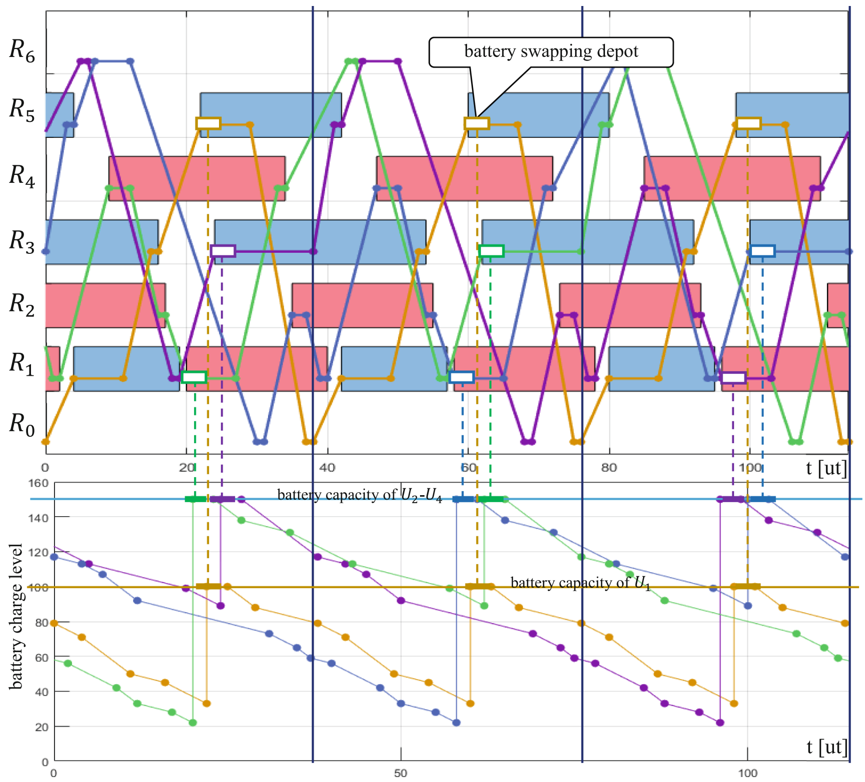


Fig. 5. The battery charge level in a system with three battery swapping depots – R_1, R_3, R_5 [source: personal collection]

assumed production takt time, subject to conditions related to the operation of a given fleet of UAVs. The solution shown in Fig. 5 is an example of optimal organization of workstation-to-workstation transport for a fleet: $U = \{U_1, U_2, U_3, U_4\}$.

This solution, however, does not take into account battery replacement (which is necessary for continued operation of an UAV) and the delays associated with it. Evaluation of whether it is possible to organize UAV battery charging in a way that will not disrupt the given schedule is an example of subproblem 1. Such an evaluation is performed by solving the appropriate problem CS_R. In particular, this problem can be considered from the perspective of the following question: *What organization of battery replacement (location of battery swapping depots and frequency of battery replacement) makes it possible to retain the desired production takt time* $TP = 38u.t.$*?*

To answer this question, one has to determine the set of operations $\mathcal{O}_{\mathcal{B}}$ after the completion of which, batteries will be swapped (at the currently occupied workstation). It is assumed that the swapping operation is carried out at least once per cycle and lasts $tw = 3$ u.t. In addition, the following parameters describing battery capacity and charge level are adopted: $PB = (100, 150, 150, 150)$; $zl = 2$ (battery consumption during the transport operation); $zs = 1$ (battery consumption during a service flight); $zp = 3$ (battery consumption during an UAV's hovering over a workstation). The operation execution schedule and the battery charge levels established by solving problem CS_R (25) are shown in Fig. 5 The calculations were done in the Oz Mozart programming environment using an Intel Core i5-3470 3.2 GHz, 8 GB RAM (calculation time, 2 s). The solution is a set of operations $\mathcal{O}_{\mathcal{B}} = \{O_1, O_4, O_6\}$.

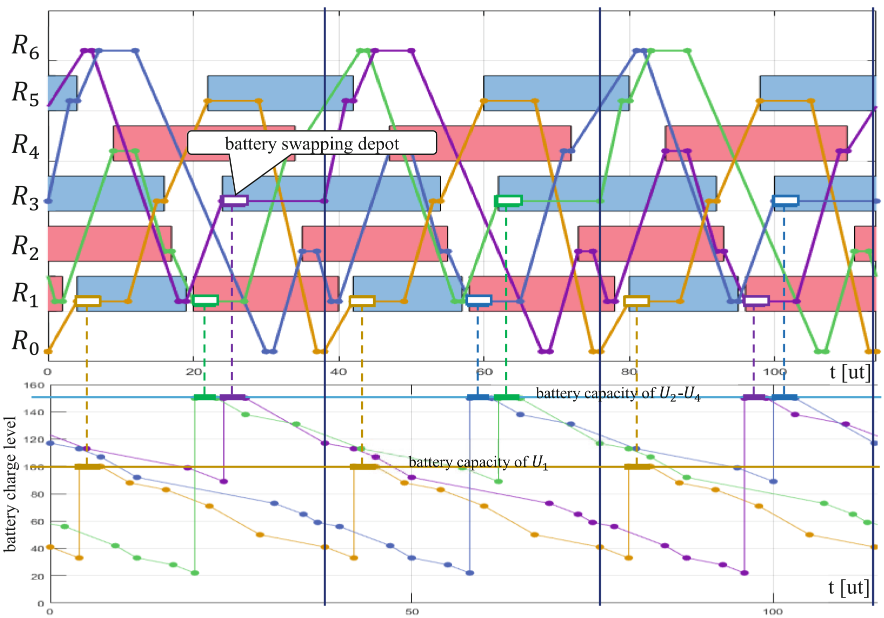


Fig. 6. The battery charge level in a system with two battery swapping depots R_1 and R_5 [source: personal collection]

The solution obtained shows that UAVs are charged cyclically at three workstations R_1, R_3, and R_5 – this way of organizing the replacement of batteries does not disrupt the current schedule and guarantees the retention of the set production takt time $TP = 38$. Once we have arrived at this point, it is only natural to ask the following question: *Can the number of battery charging depots be reduced to two?* The solution (obtained by solving problem CS_R) is shown in Fig. 7. It assumes that battery replacement takes place at workstations R_1 (twice per cycle) and R_3 (once per cycle) after the completion of operation $\mathcal{O}_{\mathcal{B}} = \{O_1, O_3, O_4\}$. Similarly to the previous solution, production takt time $TP = 38$ remains unchanged. It should be emphasized that the solution presented in Fig. 6 is optimal as regards the number of battery swapping stations.

Further reduction of the number of charging stations would require a change in system parameters, e.g. a changed battery capacity. A question that could be asked if such a problem were considered, would be the following: *Does there exist a battery capacity that will allow batteries to be replaced by using only one battery swapping station?*

As it turns out (solution CS_R (25), time 15 s), batteries with capacity $PB = (100, 200, 200, 200)$ can be replaced at only one station R_1 without the need to change the given production takt time. UAVs U_2, U_3, and U_4 are then charged alternately every third cycle, see Fig. 7. Our experiments show that in the system under consideration, a fleet of four UAVs and an appropriately organized battery replacement process enable delivery and pick-up of workpieces within the desired takt time of $TP = 38$ ut. Batteries with capacity $PB = (100, 150, 150, 150)$ enable handling of workpieces in a system that has two battery swapping stations (Fig. 6). When the battery capacity is changed to $PB = (100, 200, 200, 200)$, a single station can be used (Fig. 7).

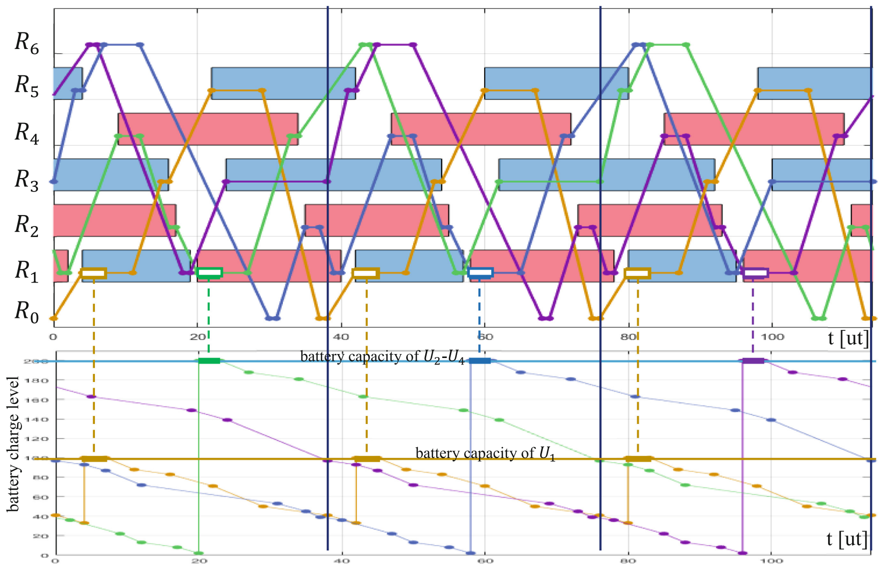


Fig. 7. The battery charge level in a system with one battery swapping depot R_1 [source: personal collection]

5 Conclusions

The possibility of flexible arrangement of collision-free flight/transport routes in 3-D space leads to new applications of UAVs in indoor settings. The flexibility of solutions which make use of UAVs is particularly manifest in situations requiring frequent changes in periodic service to and from delivery and pick-up depots (e.g. when supply chains are restructured due to changes in the organization of activities).

The present study focused on issues related to the routing of a given UAV fleet and organizing the battery replacement process (locating swapping depots and determining the frequency of battery replacement) in a way that would guarantee the maintenance of the desired production takt time. The multi-criterial and NP-hard nature of this problem narrows the set of approaches that could be used in solving it to heuristic methods. The declarative modelling framework adopted in this work allows the quick determination of a satisfactory solution for problems of a practical scale. We plan to devote future research to the problem of restructuring of delivery chains and the related issue of designing mobile battery replacement depots.

Acknowledgements. The work was carried out as part of the POIR.01.01.01-00-0485/17 project, "Development of a new type of logistic trolley and methods of collision-free and deadlock-free implementation of intralogistics processes", financed by NCBiR.

References

1. Bocewicz, G., Nielsen P., Banaszak, Z., Thibbotuwawa, A.: Routing and scheduling of unmanned aerial vehicles subject to cyclic production flow constraints. In: Proceedings of 15th International Conference on Distributed Computing and Artificial Intelligence (2018, in print)
2. Gino, J.L., Seonjin, K., Jaeyoung, C., Yibin, G., Amin, K.: Multi-UAV pre-positioning and routing for power network damage assessment. IEEE Trans. Smart Grid **PP**(99), 1 (2016)
3. Gorecki, T., Piet-Lahanier, H., Marzat, J., Balesdent, M.: Cooperative guidance of UAVs for area exploration with final target allocation. IFAC Proc. Vol. **46**(19), 260–265 (2013)
4. Guettier, C., Lucas, F.: A constraint-based approach for planning unmanned aerial vehicle activities. Constraint Satisfaction Plan. Sched. **31**(5), 486–497 (2017)
5. Guerriero, F., Surace, R., Loscri, V., Natalizio, E.: A multi-objective approach for unmanned aerial vehicle routing problem with soft time windows constraints. Appl. Math. Model. **38** (3), 839–852 (2014)
6. Hayat, S., Yanmaz, E., Muzaffar, R.: Survey on unmanned aerial vehicle networks for civil applications: a communications Viewpoint. IEEE Commun. Surv. Tutorials **18**, 1 (2016). https://doi.org/10.1109/COMST.2016.2560343
7. Ho, H.-M., Ouaknine, J.: The cyclic-routing UAV problem is PSPACE-complete. In: Pitts, A. (ed.) FoSSaCS 2015. LNCS, vol. 9034, pp. 328–342. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-46678-0_21
8. Suzuki, K.A.O., Kemper Filho, P., Morrison, J.R.: Automatic battery replacement system for UAVs: analysis and design. J. Intell. Robot Syst. **65**, 563–586 (2012). https://doi.org/10.1007/s10846-011-9616-y

9. Manyam, S.G., Rasmussen, S., Casbeer, D.W., Kalyanam, K., Manickam, S.: Multi-UAV routing for persistent intelligence surveillance and reconnaissance missions. In: 2017 International Conference on Unmanned Aircraft Systems (ICUAS), USA, pp. 573–580 (2017)
10. Michini, B., et al.: Automated battery swap and recharge to enable persistent UAV missions. In: Infotech@Aerospace 2011, American Institute of Aeronautics and Astronautics (2011)
11. Myers, D., Batta, R., Karwan, M.: A real-time network approach for including obstacles and fight dynamics in UAV route planning. J. Defense Model. Simul. Appl. Methodol. Technol. **13**, 291–306 (2016)
12. Nikhil, N.: The multiple unmanned air vehicle persistent surveillance problem: a review. Machines **2**, 13–72 (2014). https://doi.org/10.3390/machines2010013
13. Park, Y., Khosiawan, Y., Moon, I., Janardhanan, M.N., Nielsen, I.: Scheduling system for multiple unmanned aerial vehicles in indoor environments using the CSP approach. In: Czarnowski, I., Caballero, A.M., Howlett, Robert J., Jain, Lakhmi C. (eds.) Intelligent Decision Technologies 2016. SIST, vol. 56, pp. 77–87. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39630-9_7
14. Sitek, P., Wikarek, J.: A hybrid approach to the optimization of multiechelon systems. Math. Prob. Eng. **2015**, Article ID 925675, 12 pages (2015)
15. Shao, S., Guo, S., Qiu, X.: A mobile battery swapping service for electric vehicles based on a battery swapping van. Energies **10**, 1667 (2017). https://doi.org/10.3390/en10101667
16. Tan, K.C., Lee, L.H., Zhu, Q.L., Ou, K.: Heuristic methods for vehicle routing problem with time windows. Artif. Intell. Eng. **15**, 281–295 (2001)
17. Thibbotuwawa, A., Nielsen, P.: Unmanned aerial vehicle routing problems: a literature review (in print)
18. Zhihao, L., Zhong, L., Jianmai, S.: A two-echelon cooperated routing problem for a ground vehicle and its carried unmanned aerial vehicle. Sensors **17**, 1144 (2017)

Check for updates

Integral Fuzzy Power Quality Assessment for Decision Support System at Management of Power Network with Distributed Generation

Sergii Tymchuk[1], Oleksandr Miroshnyk[1], Sergii Shendryk[2], and Vira Shendryk[2](✉)

[1] Kharkiv Petro Vasilenko National Technical University of Agriculture, Kharkiv, Ukraine
stym@i.ua, omiroshnyk@ukr.net
[2] Sumy State University, Sumy, Ukraine
{s.shendryk, v.shendryk}@cs.sumdu.edu.ua

Abstract. This paper is devoted to the development of scientific and methodological foundations of improvement the information support of decision at management of power network with distributed generation. It is proposed to consider the power quality index as the main criterion of management. Using the theory of fuzzy sets, the assessment of the conformity of power quality indicators to electric energy quality limits is done. A method for estimating the quality of electrical energy is proposed which represents the measured histogram as a fuzzy representation indicator of quality of electric energy in the form of fuzzy sets with a step membership function. The method of integral evaluation of electrical energy quality for different types of load is developed. The presented method allows to formulate rules for managing the operating modes of the distributed electrical network by the decision support system.

Keywords: Distributed generation · Decision support system
Power quality index · Fuzzy sets · Integral index

1 Introduction

The formation of a distributed generation infrastructure is one of the perspective directions for the development of the world power industry. This concept provides for a variety of consumers that generate electrical energy for their own needs and direct excess electricity to the general network. The distributed generation of electricity is characterized by low maintenance costs, low level of environmental pollution and high efficiency [1].

The distributed generation infrastructure successful functioning depends on the efficient operation of the generation sources, the coordination of consumption and reliable integration into the external network. It can be achieved by increasing the number of interrelationships in the electrical network. In turn, such infrastructure becomes more complicated for average consumers to manage it, as it becomes more difficult to monitor and predict the work of the electrical networks. It happens, because the system is characterized by a rapid change of operating modes, depending on

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 88–97, 2018.
https://doi.org/10.1007/978-3-319-99972-2_7

external conditions. The process of making decision in case of distributed network management is characterized by big amount of discrete variables that have impact on overall system. Operativity of decision-making process plays an important role in the management process. It can be achieved by means of decision support systems. So it is necessary to implement information technologies in the management process of distributed power network operating modes. This solution will allow consumers to make decisions which provide generation efficiency and consumption of electricity [2].

The electric power quality can be considered as the main performance criteria of the distributed network.

But in fact the term «electric power quality» is fuzzy. So, in order to assess the electric power quality, a system of indexes and limits was introduced [3]. The available limits of quality indexes are presented as intervals, and the quality indexes are calculated by complex procedure based on multiple measurements. It is evident that this leads to some uncertainty when determining the index as well as assessing its conformity with the limits. In normative documents, this uncertainty is artificially determined by the use of mathematical statistic methods and complex inequalities. This approach does not permit to assess the violation degree of quality limits, to monitor the trend of quality index values and make integral estimate the electric power quality.

2 Literature Review and Formulation of Problem

The assessment of Power Quality Index (PQI) and disclosure of information uncertainty during the analysis of the PQI are considered today as unrelated tasks. The deterministic approach is widely used for calculations. As to the measurements, according to the international requirements of metrology and standardization, it is recommended to consider the main quality assessment of measurements as uncertain [4]. If the process of the PQI measurements contain the substantial stake of uncertainty beforehand, there are two ways to overcome it: to use expert knowledge to make decision under uncertainty conditions or to use the fuzzy set theory for assessment of measurement uncertainty [5, 6]. In [4], it is recommended to express uncertainty of operation parameters by means of symmetric boundaries or mean square deviation. In a case of the mean square deviation, the probabilistic assessment of measurement set are used, and in a case of symmetrical boundaries, it is possible to use subjective knowledge which can be easily formalized using the fuzzy set theory.

A fuzzy approach is also used in operating management systems of hybrid electric grid. In most cases, these are systems for voltage [7] or power control [8]. Such approach is also used as a concept for assessment of damage caused by poor quality electric energy [9].

In [10], it is recommended to use the fuzzy set theory when assessing the PQI in the decision support systems since the fuzzy representation describes an object in simpler way. This speeds up the decision-making process. If the distribution of the measurement results is assumed to be symmetric and unimodal, so such results can be represented as triangular fuzzy numbers [7].

Also, it is proposed to use the fuzzy numbers approach in calculation of uncertainty for indirect measurement with confidence interval if the error equation cannot be linearized [5, 12]. However, if the distribution is asymmetric, the results will not be acceptable. The authors recommend making the evaluation of distributions in responsible cases.

In [13], it is suggested to use a fuzzy number with a triangular membership function to assess the PQI. This approach is valid when the set of measurement results has signs of unimodality and symmetry.

All considered approaches require a preliminary analysis of the measurement results to determine whether the means of evaluation indeterminate forms were used correctly or not. This leads to complexity in the hardware implementation. Meanwhile, some instruments for measuring the PQI can present the results in the form of a histogram. After normalization such histogram can be regarded as a fuzzy representation of the PQI in the form of a fuzzy set with a step membership function enveloping the histogram. Of course, in this case there is no need to impose restrictions on unimodality and symmetry and thus exclude the stage of preliminary data analysis.

3 Definition of the Goal

The purpose of this article is to determine the performance criterion of the distributed network which is power quality index in a fuzzy form for use it under conditions of uncertainty information in a decision support system. In order to achieve the purpose in view, it is necessary to solve the following tasks:

- conversion of the measurement results of the power quality index into a fuzzy form;
- fuzzy representation of power quality limits;
- fuzzy conformity assessment of power quality index to the relevant limits;
- formation of an integral fuzzy concept of power quality.

4 Conversion of Measurement Results of Power Quality Index into Fuzzy Form

The *PQI* as voltage fluctuation, non-sinusoidal voltage, voltage unbalance, frequency deviation, etc. are measured within 24 h. A set of values of ΔPQI of dimension N_{Δ} is formed during that period of time (Fig. 1).

The instrument forms a histogram using the given set of measurements. The range $\{PQI_{min}, PQI_{max}\}$ is divided into n of equal intervals d_i, and the histogram for frequencies of getting of the *PQI* measurements into specified intervals is built. After normalization we obtain the histogram (Fig. 2).

We form a fuzzy set of the PQI based on the obtained histogram. This set is within the range $\{PQI_{min}, PQI_{max}\}$. The membership function is a step function (μ) enveloping the histogram (Fig. 3).

As we can see, this representation of the quality index does not require conversion the fuzzy set into any particular case of well-studied types of fuzzy sets. Consequently, there is no need to make a significant upgrade of the measurement instrument software.

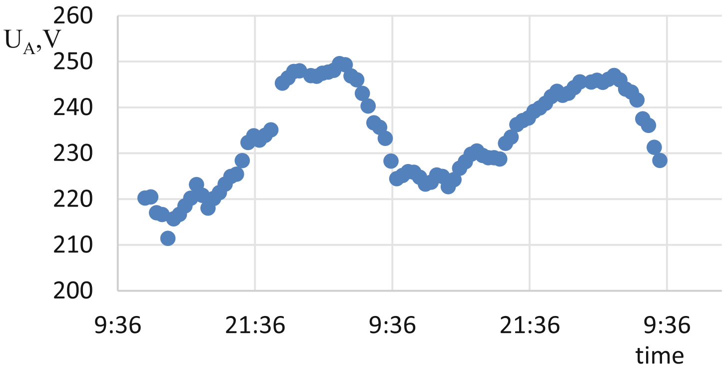


Fig. 1. Time series of measurements (on Example of Phase A Voltage)

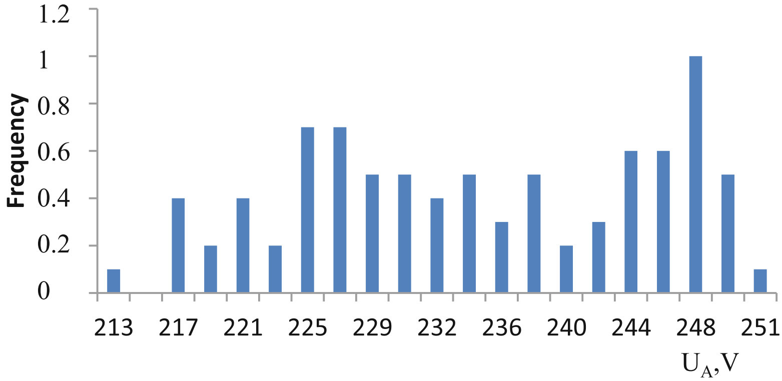


Fig. 2. Normalized histogram of measurements (on Example of Phase A Voltage)

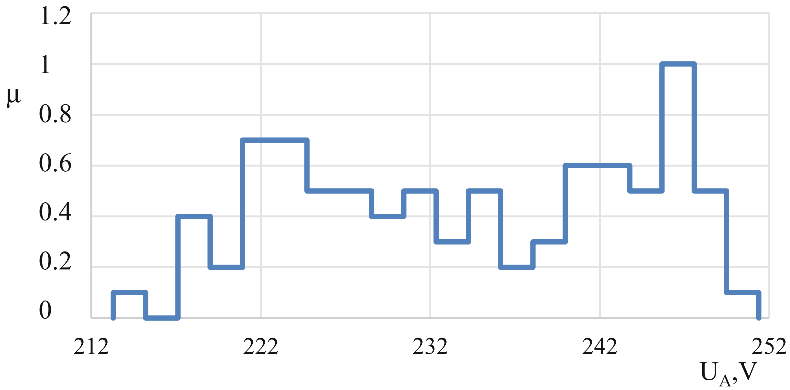


Fig. 3. Power quality index in fuzzy form

However the peculiarity in the measurement of quality index within a particular power network is still presented.

5 Fuzzy Representation of Power Quality Limits

Power quality limits (*PQL*) are defined in the form of intervals of normally admissible and maximum permissible values. That is, the limit definition itself contains an element of uncertainty. It is because there is a necessity to define if the PQI meets the quality limits within the interval between the normally admissible and maximum permissible values.

For the decision making system it is important not only to make the record when the quality index exceeds the limits but also it is necessary to prevent the risk of such situation.

These limits can be represented by a fuzzy set that is a fuzzy interval with a trapezoidal membership function.

$$\mu_{PQL} = \max\left\{0, min\left\{1, \frac{PQL - PQL_{min}}{PQL_{m1} - PQL_{min}}, \frac{PQL_{max} - PQL}{PQL_{max} - PQL_{m2}}\right\}\right\} \tag{1}$$

The values of PQL_{min}, PQL_{max} in Eq. (1) are specified in the normative documents (such as [3]). The values of PQL_{ml}, PQL_{m2} are received from expert estimation or empirical study. The system sensitivity to the PQI variation depends on the value of these parameters.

This kind of membership function allows us to introduce some order relation in the assessment of the *PQI*. Since the *PQI* approaches the maximum permissible values, the membership function approaches zero.

The maximum normal and maximum permissible values are only defined for some *PQL*. In this case, Eq. (1) is somewhat simplified (PQL_{min}= PQL_{m1} = 0).

6 Conformity Assessment of Unified Power Quality Index to Established Limits in Fuzzy Form

The conformity degree of fuzzy values of the *PQI* to fuzzy *PQL* can be evaluated by the membership function of their intersection.

$$\mu_{PQ} = \min\left(\mu_{PQL}, \mu_{PQI}\right) \tag{2}$$

In this case the conformity degree is preferred to express as number rather than function. So we propose the following approach.

The intersection of fuzzy sets can be numerically estimated by the area of intersection of two shape (*S*). The first one is a shape bounded by the abscissa axis and the membership function of the *PQI* (S_{PQI}) [4]. The second one is a shape bounded by the abscissa axis and the membership function of *PQL* (S_{PQL}) (Fig. 4).

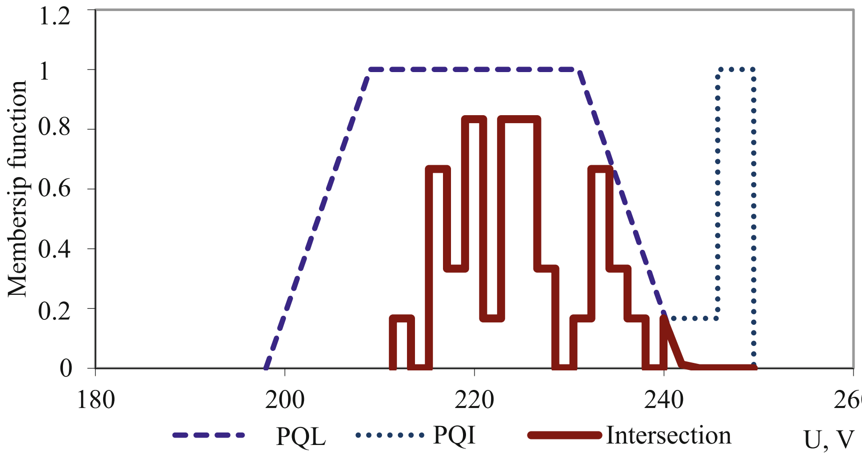


Fig. 4. The intersection of fuzzy sets

Then membership function of conformity of the fuzzy PQI to the fuzzy limits of power quality (μ_{PQ}) can be represented as

$$\mu_{PQ} = S/S_{PQI}, \tag{3}$$

$$S_{PQI} = \sum\nolimits_{i=1}^{n} d_i \mu_i, \tag{4}$$

$$S = \sum\nolimits_{i=1}^{n} d_i min(\mu_i, \mu_{PQL})$$

where d_i is the width of region i, μ_i is the value of the *PQI* membership function on the region i (Fig. 3). Let's consider two cases which show the difference between fuzzy and deterministic approaches. In the first case the measurements of the voltage in the phase A (Fig. 1) is the following: 15% of the values exceeds the upper limit. According to the limits [3] the *PQI* for voltage deviation in phase A does not confirm the limits. Fuzzy assessment of the conformity of this parameter to the limit (Fig. 4) was 0.699. This assessment versus the deterministic one presents the degree of conformity of this parameter to the limit. In the second case the voltage measurements in phase A show that there is no violation of the limits. So, the PQI for voltage deviation in the phase A conforms the limits. The fuzzy assessment of this parameter (Fig. 5) was 0.801. It means that the value of this parameter approaches the limits. This data can be used in the decision support system for preventive actions regarding normalization of voltage deviation in the phase A. Also, the fuzzy assessment provides the possibility to monitor and quickly assessment the efficiency of the accepted decisions aimed to improve the electric power quality even if quality indexes conform the limits.

The considered cases show that the fuzzy assessment of the quality indexes is more informative in terms of intelligent decision support system.

There are the number of indexes characterizing various aspects of the power quality. Most of them can be presented in the fuzzy form according to the proposed

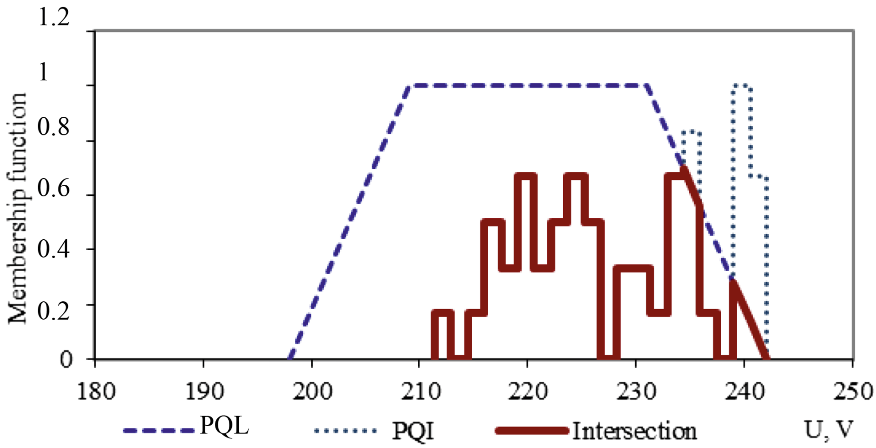


Fig. 5. The intersection of fuzzy sets (Case 2)

method. But there are peculiarities for some indexes. For example, since the normative document does not include the number of measurements of the voltage fluctuation range δU_t, then a single measurement can be considered as a singleton. Similarly, each flicker dose value has an independent value and can be represented as a singleton. Their membership functions are:

$$\mu_{\delta U_t}(\delta U_t) = 1, \mu_{P_{St}} = 1, \mu_{P_{Lt}} = 1. \tag{5}$$

7 Formation of Integral Fuzzy Concept of Power Quality

The fuzzy representation of the conformity of the quality indexes to the accepted limits allows us to form integral fuzzy concepts of the power quality. This is because the fuzzy set operations are definitely projected on membership function operations. In this case, the physical nature of the quality index is of no consequence.

The simplest approach to the formation of an integral concept of the quality based on the logical operation of fuzzy sets intersection was given by the authors in [9]. The fuzzy concept of the quality is formed as follows

$$PQ = \bigcap_{j=1}^{N_{PQ}} PQ_j, \tag{6}$$

where N_{PQ} is the number of quality indexes considered.

In this case, the power quality level is numerically represented by the membership function of the fuzzy set of *PQ* which for the intersection operation will be as follows

$$\mu_{PQ} = \min\left(\mu_{PQj}\right). \tag{7}$$

Then we can consider μ_{PQ} as a generalized index that assess the power quality as a number from the range [0, 1].

It should be noted that the number of indexes taken into account should be correlated with the needs of consumers. Different types of loads are critical for different power quality indexes. Therefore, it is possible to form integrated generalized quality indexes for power networks with predominance of some type of load using the described approach.

For example:

– for electromotor load

$$\mu_{PQ} = \min\left(\mu_{\delta U_y}, \mu_{K_U}, \mu_{K_{U(n)}}, \mu_{K_{2U}}\mu_{K_{0U}}\mu_{\Delta f}, \mu_{\Delta t_n}, \mu_{K_{pU}}\right); \tag{8}$$

– for lighting load

$$\mu_{PQ} = \min\left(\mu_{\delta U_y}, \mu_{\delta U_t}, \mu_{P_t}, \mu_{K_{0U}}, \mu_{\Delta t_n}, \mu_{K_{impU}}\right); \tag{9}$$

– for devices with microprocessor control units

$$\mu_{PQ} = \min\left(\mu_{\delta U_t}, \mu_{P_t}, \mu_{K_U}, \mu_{K_{U(n)}}\right). \tag{10}$$

In Eqs. (8)–(10): δU_y is the steady-state voltage deviation; δU_t is the range of voltage fluctuation; P_t is the flicker dose; $K_{U(n)}$ is the coefficient of the n-th harmonic component of the voltage; K_{2U} the voltage unbalance ratio at reverse sequence; K_{0U} is voltage unbalance ratio at zero sequence; Δf is frequency deviation; Δ_{tn} is the voltage dip interval; K_{pU} is the temporary overvoltage factor; K_{impU} is the voltage impulse.

It can be easily seen that the proposed approach does not contain complex mathematical calculations and can be easily implemented in the software of decision support system.

In Eqs. (7)–(10), the generalized power quality indexes can take values from the range [0, 1]. At the same time, if follow the requirements of existing limits for assessing the power quality, then μ_{PQ} values which are not equaled to1 are definitely qualified as the lack of the required power quality. With a deeper implementation of the fuzzy approach in assessing the power quality, it is possible to avoid such a rigid differentiation derived from the deterministic approach. For example, you can enter the permissible values of fuzzy generalized quality indexes for each type of load. However, this provision requires a special research and can serve as a basis for reviewing the existing approach for assessing the power quality.

The developed methodology allows us to monitor the power quality variation even if the main indexes do not exceed the limits of acceptable values, to analyze the dynamics of the PQI variation and determine the prevention measures for the normalization of the PQ, i.e. it can be considered as the main method for managing the operation mode of the distributed network.

8 Conclusions

The developed fuzzification method of quality indexes based on the histogram of measurement set makes it possible to perform the fuzzy assessment of the power quality indexes.

The presented method of fuzzy conformity assessment of the quality indexes to the established fuzzy limits allows us to form various integral quality indexes which consider load type in the power networks.

The proposed method allows us not only to assess the conformity degree of quality indexes with the established limits, but also to monitor the power quality variation even if the main indexes do not exceed the limits of acceptable values, to formulate rules for managing of the operating modes of the distributed electrical network by the decision support system.

The methodology of decision-making based on the fuzzy assessment of the quality of electricity (defuzzification) is the subject of further research and has not been considered in this article.

References

1. Pakštas, A., Shulyma, O., Shendryk, V.: On defining and assessing of the energy balance and operational logic within hybrid renewable energy systems. Commun. Comput. Inf. Sci. **639**, 151–162 (2016)
2. Shulyma, O., Shendryk, V., Parfenenko, Y., Shendryk, S.: The model for decision support on design of the hybrid renewable energy system. In: Proceedings of the 2017 IEEE 9th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2017, pp. 47–50 (2017)
3. EN 50160:2010 Voltage Characteristics of electricity supplied by public distribution networks
4. Guide to the Expression of Uncertainty in Measurement: First Edition, p. 101. ISO, Switzerland (1993)
5. Mauris, G., Lassere, V., Foulley, L.: A fuzzy approach for the expression of uncertainty in measurement. Measurement **29**, 109–121 (2001)
6. Mauris, G., Berrah, L., Foulloy, L., Haurat, A.: Fuzzy handling of measurement errors in instrumentation. IEEE Trans. Measur. **1**(49), 43–58 (2000)
7. Canha, L.N., Pereira, P.R., Milbradt, R., Da Rosa Abaide, A., Schmitt, K.E.K., De Abreu Antunes, M.: Intelligent voltage regulator to distributed voltage control in smart grids. In: 52nd International Universities Power Engineering Conference, UPEC 2017, 2017 January, pp. 1–6 (2017). https://doi.org/10.1109/upec.2017.8231977
8. Arcos-Aviles, D., Pascual, J., Marroyo, L., Sanchis, P., Guinjoan, F.: Fuzzy logic-based energy management system design for residential grid-connected microgrids. IEEE Trans. Smart Grid **9**(2), 530–543 (2018). https://doi.org/10.1109/TSG.2016.2555245

9. Liu, X.N., Wei, J., Ye, S.Y., Chen, B., Long, C.: Research on uncertainty evaluation measure and method of voltage sag severity. In: IOP Conference Series: Earth and Environmental Science, vol. 108, no. 5 (2018). https://doi.org/10.1088/1755-1315/108/5/052098
10. Miroshnik, A.A., Tymchuk, S.A.: Uniform distribution of loads in the electric system 0,38/0,22 kV using genetic algorithms. Tech. Electrodynamics **4**, 67–73 (2013)
11. Piegat, A.: Fuzzy Modeling and Control. Phisical-Verl, Heidelberg, New York (2001). 728
12. Tymchuk, S.A., Miroshnyk, A.A.: Quality assessment of power in distribution networks 0.38/0.22 kV in the fuzzy form. In: Global Science and Innovation. Materials of the II International Scientific Conference, vol. II, Chicago, USA, pp. 288–299 (2014)
13. Tymchuk, S.A., Miroshnyk, A.A.: Assess electricity quality by means of fuzzy generalized index. East Eur. J. Adv. Technol. **3/4**(75), 26–31 (2015)

Development of Models for Computer Systems of Processing Information and Control for Tasks of Ergonomic Improvements

Evgeniy Lavrov[1] and Nadiia Pasko[2(✉)]

[1] Sumy State University, Sumy, Ukraine
prof_lavrov@hotmail.com
[2] Sumy National Agrarian University, Sumy, Ukraine
senabor64@ukr.net

Abstract. The questions of search of ergonomic reserves of efficiency of computer systems of processing information and control are considered. A set of models of a computer system of processing information and control, describing it in the necessary sections, was developed. The results can be useful in design of information provision for Decision Support Systems, devoted to questions of programs ergonomic quality of automated systems.

Keywords: Ergonomics · Human factor · Computer system · System analysis Component model · Morphological model

1 Introduction

In modern distributed automated systems that process information and manage objects of a different nature, the role of the human-operator has changed significantly. Complex automation of production not only does not relieve a person from the operational management of technological processes, but also complicates his activities. In this regard, many researchers in their work set the goal to take into account the "human factor" as much as possible [1–3].

Taking into account the "human factor" becomes one of the most important questions in design and employment of automatized systems [4–8]. The effectiveness of complex computer systems of processing information and control (SPIC), which belong to the class of ergotechnical systems (ETS) [9] essentially depends on the thoroughness of ergonomic issues [10–13]. To address such concerns, the models of searching for ergonomic reserves to increase efficiently of SPIC are being developed [13–15] including the methods of fuzzy logic [16–18]. Introduction of such models is constrained by the lack of information support, which makes it possible to systematically describe the subject area of activity of decision-makers and ensure optimal design and management of reliable initial data.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 98–109, 2018.
https://doi.org/10.1007/978-3-319-99972-2_8

2 Statement of the Task

To develop methodology for systemic-ergonomic analysis, which can be used as a basis for information support of decision making assistance systems (DSS) on the issues of ergonomic quality assurance of the SPIC (for the operator-supervisor, the ergonomics department staff, managers and other employees responsible for elaboration of the issues of account of the "human factor").

3 Results

3.1 Models for Information Support of DSS on Issues of Improving the Ergonomic Quality of SPIC

We construct system models using an approach to the unified representation of information about ETS objects in the form of a list of knowledge and data bases, as described in [9].

Analysis of the list of information about SPIC, necessary for solving the problems of ergonomic support, allows us to conclude that this information can be specified using two classes of structures, namely component and morphological.

Component structures are introduced to identify the entities needed to describe the system.

Morphological structures are introduced to define connections of a different nature between entities identified in component structures. To construct a model describing individual algorithms for each operator to perform each type of application, the apparatus of functional networks of the functional-structural theory of the ETS of the school of prof. A.I. Gubinsky is used [9]. Then the complex of system models *MMS* of an information model for the operator-supervisor is represented by the scheme shown in Fig. 1 and the structural formula (1):

$$\begin{aligned} MMS = \ < & CSs, CFs, CRs, CEs, Err, Rs, Ft, OpFt, \\ & VPo, FKv, Mpl, VCo, M \text{ Pr } oekt > \end{aligned} \tag{1}$$

Description of the accepted notations is shown in Fig. 1.

3.2 Contents of the Models

Here we present several examples of the developed models.

Component Model of System, CSs. The component-system structure reflects the location of this SPIC among the surrounding systems and the composition of its subsystems:

$$CSs \ = \ <\{LETS_l\}\,|\,l = 1, 2, \ldots L_0 > , \tag{2}$$

where $LETS_l$ – the l-th local ETS;
L_0 – the number of local ETS systems

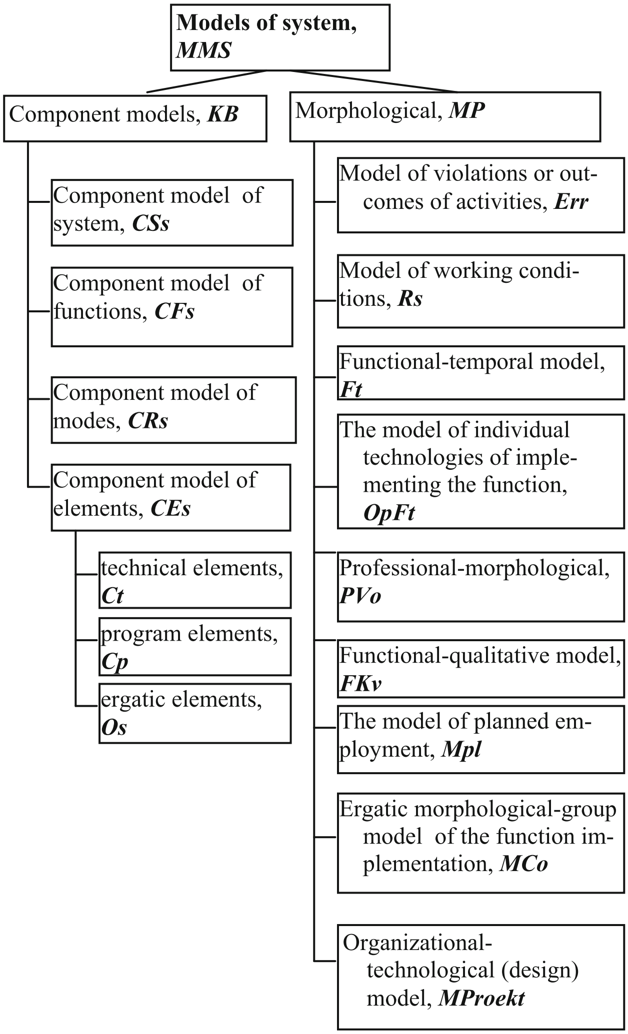


Fig. 1. Structure of the information model for computer systems of processing information and control

Component-Functional Structure, CFs. The model represents information about the requirements for the system under study in terms of the set of realized planned functions of each LETS:

$$CFs \;=\; <\{Fpl_i; \{LETS_{l_i}\} | l_i \in \{1, 2, \ldots L_0\}\} \big| i = 1, 2, \ldots, N_{Fpl} > \qquad (3)$$

where $LETS_{li}$– l_i-th local ETS;
L_0 – number of local ETS systems;
Fpl_i – the i-th planned function;
N_{Fpl} – the number of planned functions

Model of Violations or Outcomes of Activities, Err. The model describes possible violations when implementing functions (application) from a set given by a component-functional structure:

$$Err = <\{Fpl_i; \{Er_{ij}; U_{ij}\} \big| j = 1, 2, \ldots ner_i\} \big| i = 1, 2, \ldots, N_{Fpl} > , \tag{4}$$

where Fpl_i – the i-th planned function;

N_{Fpl} – the number of planned functions;

Er_{ij} – the j-th violation (error), which can be tolerated in the implementation of the i-th function;

U_{ij} – the damage caused by performing the i-th function with j-th violation;

ner_i – the number of possible violations in the implementation of the i-th planned function

Model of Working Conditions, Rs. Displays for each workplace the values of the sanitary, hygienic and psychophysiological factors affecting the working conditions:

$$\begin{aligned} Rs = \; & <\{RM_k; OP_k; KT_k; IBO_k; \\ & \{TFak_{kj}; NFak_{kj}; ZFak_{kj}\} | j = 1, 2, \ldots nf_k\} | i = 1, 2, \ldots K_0 > , \end{aligned} \tag{5}$$

where RM_k – the identification of the k-th workplace;

OP_k – the k-th operator;

KT_k – the category of gravity for the workplace RM_k;

IBO_k – the integral score in points for the workplace RM_k;

nf_k – the number of influential factors for the workplace RM_k;

$TFak_{kj}$ – the type of j factor for the workplace RM_k;

$NFak_{kj}$ – the name of j factor for the workplace RM_k;

$ZFak_{kj}$ – the value of the j-th influencing factor for the workplace RM_k;

K_0 – the number of workplace of system

Functional-Temporal Model of Performing Functions, Ft. The model describes typical algorithms for performing functions (application) defined by a component-functional structure. There are several possible options for organizing activities in the implementation of the function. The model is given in the form of a formal model of a functional network:

$$Ft = <\{Fpl_i; \{Var_{ij}, Mfs_{ij}\} | j = 1, 2, \ldots nv_i\} \big| i = 1, 2, \ldots, N_{Fpl} > , \tag{6}$$

where Fpl_i – the i-th planned function;

N_{Fpl} – the number of planned functions;

Var_{ij} – the j-th variant of the organization of activities in the realization of i-th function;

Mfs_{ij} – the j-th formal model of the functional network of the algorithm for the implementation of the i-th function;
nv_i – the number of variants for organizing activities in the implementation of the i-th function

The Model of Individual Technologies of Implementing the Function, OpFt. The model describes typical technologies for the implementation of functions (application) that characterize the organization of the activity of system operators in the form of a formal model of a functional network. The model describes typical algorithms for performing functions (application) defined by a component-functional structure. There are several possible options for organizing activities in the implementation of the function:

$$\begin{aligned} OpFt \; = \; & < \{Fpl_i; OP_{ik}; \{Var_{ij_k}, Mfs_{ij_k}\} \\ & |j_k \in \{1, 2, \ldots nv_i\}\} | k \in \{1, 2, \ldots, K_0\} \left| i = 1, 2, \ldots, N_{Fpl} > \right., \end{aligned} \tag{7}$$

where Fpl_i – the i-th planned function;
OP_{ik} – the k-th operator that implements the i-th planned function;
VAR_{ijk} – the j-th variant of the organization of activity by the k-th operator when implementing the i-th function;
Mfs_{ijk} – the j-th formal model of the functional network of the algorithm for executing the i-th function by the k-th operator;
nv_i – the number of variants for organizing activities in the implementation of the i-th function.;
N_{Fpl} – the number of planned functions

Functional-Qualitative Model, FKv. The model defines the probabilistic characterization of processes of occurrence and elimination of errors of each type to be taken into account when executing the function (application) from the set, which is specified by the component-functional structure. The set of operations of the process of performing functions is determined by the functional-temporal structure, taking into account the values of factors of the structural feature of the workplace (component-element structure), working conditions at the workplace and professional characteristics of the operator. In the case of absence of data on a particular operator, the model specifies the probabilistic characteristics of the average operator:

$$FKv = \begin{cases} FKvo, & \text{if the data on operators of system are known} \\ FKvs, & \text{if there is no data on operators of system} \end{cases} \tag{8}$$

$$FKvo = <\{Fpl_i; OP_{ik}; \{Var_{ij_k}; \{Opr_{ij_kl}; \{PK^{er_i}_{ij_kl_m} \Big| m = 1,2,\ldots np^l_{j_k}; \Big| er_i = 1,2,\ldots,ER_i\} \Big| l = 1,2,\ldots,ko^i_{j_k}\} \Big| j_k \in \{1,2,\ldots,nv_i\}\} RM_{ik}; KT_{ik}\} | k \in \{1,2,\ldots K_0\}; | i = 1,2,\ldots,N_{Fpl} > ;$$

$$FKvs = <\{Fpl_i; \{Var_{ij}; \{Opr_{ijl}; \{PK_{ijl_m} \Big| m = 1,2,\ldots np^l_{j_k}\} \Big| l = 1,2,\ldots ko^i_j\} | j = 1,2,\ldots nv_i\}; RM_{ik}; KT_{ik}\} | i = 1,2,\ldots,N_{Fpl} > ;$$

where Fpl_i – the i-th planned function;

N_{Fpl}	– the number of planned functions;
OP_{ik}	– the k-th operator that implements the i-th planned function;
$Var_{ijk}(Var_{ij})$	– the j-th variant of the organization of activity by the k-th operator (by the average operator) in the implementation of the i-th function;
RM_k	– the identification of the k-th workplace;
KT_k	– the category of gravity for the workplace RM_k;
$Opr_{ijkl}(Opr_{ijl})$	– The l-th operation performed by the k-th (average) operator when implementing the i-th planned function in the j-th way;
PK^{eri}_{ijklm}	– The m-th quality index, which takes into account the error of er_i-th type, with which the k-th operator performs the l-th operation when implementing the i-th planned function in the j-th mode;
PK_{ijlm}	– The m-th quality index with which the average operator performs the l-th operation when implementing the i-th planned function in the j-th mode;
ER_i	– errors of different types that can be admitted in the implementation of the i-th function

The Model of Planned Employment, Mpl. The model shows the state of employment of operators by performing routine functions. The set of functions is determined by the component-functional structure of the system. For each operator, the model specifies the start time and the completion time for each function assigned to the operator, the possibility of interrupting the function, the time of the scheduled break, etc.:

$$Mpl = <\{RM_k; OP_k; \{PpN_{kl}; PpK_{kl}\} | l = 1,2,\ldots,n_k; \{Fpl_{ik}; TN_{ik}; TK_{ik}TP_{ik}; Vyp_{ik}; TF_{ik}; PR_{ik}\} | i = 1,2,\ldots,m_k\} | k = 1,2,\ldots,K_0 > , \quad (9)$$

where RM_k– the identification of the k-th workplace;

OP_k	– the k-th operator;
PpN_{kl}	– the beginning of the l-th scheduled break of the k-th operator;
PpK_{kl}	– the end of the l-th scheduled break of the k-th operator;
n_k	– the number of scheduled breaks of the k-th operator;
Fpl_{ik}	– the i-th planned function of the k-th operator;
TN_{ik},	– the time of start for executing the i-th planned function of the k-th operator;
TK_{ik}	– the completion time of the i-th planned function of the k-th operator;

TP_{ik}, – the time of interruption of execution of i-th planned function of the k-th operator;
TF_{ik} – the actual completion time of the i-th planned function of the k-th operator;
Vyp_{ik} – the mark on the fulfillment of the i-th planned function of the k-th operator;
PR_{ik} – the priority of the i-th planned function of the k-th operator. The priority of the function is determined by a ten-point scale;
m_k – the number of planned functions of the k-th operator;
K_0 – the number of workplace of system

Ergatic Morphological-Group Model of the Function Implementation, MCo. Describes for each function (application) from the set, defined by a certain component-functional structure, possibility (impossibility) of organization of group activities and pairwise compatibility of the operators during implementation of the function (in case of admissibility for group activities):

$$MCo \;=\; <Fpl_i; pr_i; [\{C^i_{kl}\}|k = 1, 2, .., K_0; l = 1, 2, \ldots, K_0\}]|i - 1, 2, \ldots, N_{Fpl} > , \tag{10}$$

where Fpl_i – the i-th planned function;
pr_i – a sign that determines the possibility or impossibility of organizing a group activity;
$\{C^i_{kl}\}$ – matrix, which defines pairwise compatibility of the functioning of operators OP_k and OP_l during realization of the i-th planned function. Each element of the matrix is defined by the following formula:

$$C^l_{kl} = \begin{cases} 1, & \text{if operators can be jointly involved} \\ 0, & \text{if operators can not be worked together} \end{cases} \tag{11}$$

The Organizational and Technological (Design) Model, MProekt. The model displays the result of solving the problem of prescribing the function for the operators and contains:

- name (identification) of the function;
- description of the technology implementation of the function (functional network of the functional-temporal model or from the model of individual technologies);
- type of organization of activities: individual or group;
- in the case of individual activities - identification of the operator who is entrusted with the execution of the application. In the case of group activities - assignment of a responsible operator for each operation.
- Prediction of the results of solution: – mathematical expectation of execution time; – variance of the execution time; – probability of error: 1-th type; 2-th type; …n-th type; – the amount of possible damage from violations of different types.

$$M\,\mathrm{Pr}\,oekt \;=\; <\{Fpl_{i_0}; Var_{i_0j_0}; MFs_{i_0j_0}; pr_{i_0j_0}; [OP_{i_0k_0}]; [\{Opr_{i_0j_0l}; OP_{l_m}\}|m \in \{1,2,\ldots,K_0\}; \\ |l = 1,2,\ldots ko^{j_0}_{i_0}]|k_0 \in \{1,2,\ldots,K_0\}|j_0 \in \{1,2,\ldots nv_{i_0}\}|i_0 \in \{1,2,\ldots,N_{Fpl}\}\} >, \tag{12}$$

where Fpl_{io} – the i_O-th function, received by application for execution;

Var_{iojo} – selected the j_o-th variant of the organization of activities in the implementation of the i_o-th function which has been received with the application for execution;

Mfs_{iojo} – selected the j_o-th formal model of the functional network of algorithm implementation of the i_o-th function;

OP_{ioko} – selected the k_o-th operator for the implementation of the application of the i_o-th function in the case of the impossibility of group activity;

$\{Opr_{iojol}\}$ – the set of operations of the activity algorithm for the selected j_o-th variant of the organization of activities for implementation of the i_o-th function which has been received with the application for execution;

ko^{jo}_{io} – the number of operations of the activity algorithm for the selected j_o-th variant of the organization of activities for implementation of the i_o-th function which has been received with the application for execution;

nv_{jo} – the number of options for organizing activities in the implementation of the i_o- function;

$\{OP_{lm}\}$ – the group of operators assigned to execute l_m-th operations j_o-th variant of the organization of activities in the implementation of the i_o-th function which has been received with the application for execution. To implement the function, group activities are allowed

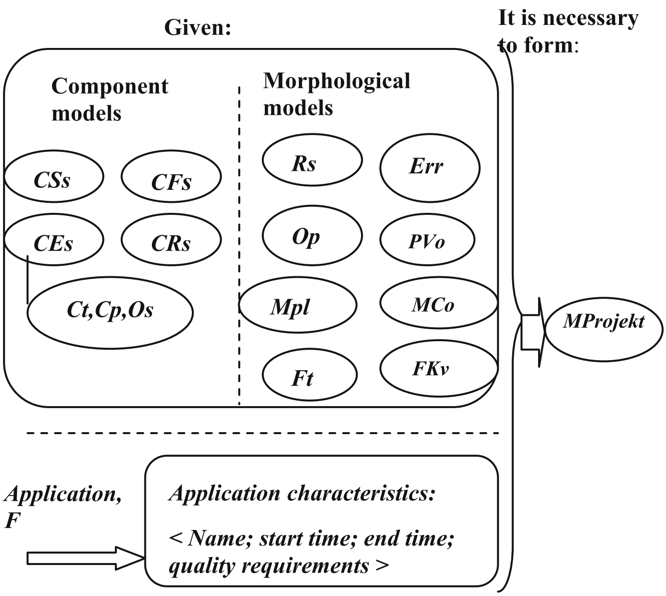


Fig. 2. Illustration for the meaningful establishment of the task of assigning functions to operators using the technology of component and morphological analysis of SPIC

The symbols “[]” in the models mean that optional elements are enclosed in parentheses.

For example, this model (structure) can demonstrate the “relationship” of different models in the process of solving tasks for ensuring ergonomic SPIC on a model example tasks of ergonomics, which is distribution of functions between operators.

It is obvious that the task of distributing functions can be represented as a task of forming a new organizational and technological structure which satisfies the requirements to the quality of the application and a variety of ergonomic standards and requirements (Fig. 2).

Description of the accepted notations on Fig. 2 is shown in Fig. 1.

4 Approbation. Using of Results and Their Practical Significance

The approach was used in the development of the modeling qualimetric complex of ergotechnical systems [15, 19–24].

The results were used in solving the following problems:

- Development of information models for a human operator [21, 22];
- Development of algorithms for operators [23];
- Distribution of functions between operators [19, 20];
- Distribution of functions between a human and automation devices [15].

The results are used in the design and utilization of systems for various purposes (automated control systems, payment centers, contact centers, e-learning systems, etc.).

Description of one of the many computer experiments that we conducted is given below. Let us consider the problem of the distribution of functions between operators of an automated system [24].

When examining various ways of assigning requests to performers, we considered a limited number of strategies [24]:

- S0 - random generation of the option;
- S1 - random selection of free operators;
- S2 - random selection of the most prepared operators;
- S3 - random selection of the most prepared free operators with acceptable working conditions in the workplace;
- S * - the option chosen by the computer system, which is constructed using the models described in this paper.

The average probability of error-free and timely execution for request (for simplicity of presentation one type of possible requests is shown), depending on the chosen strategy (level of use of the automated system), is shown in Fig. 3.

In this case, the choice of the optimal strategy is realized using the optimization model [24]. However, the practical implementation of the problem is possible only when using the complex of models described in this paper. Thus, numerical modeling confirms the constructiveness of our approach.

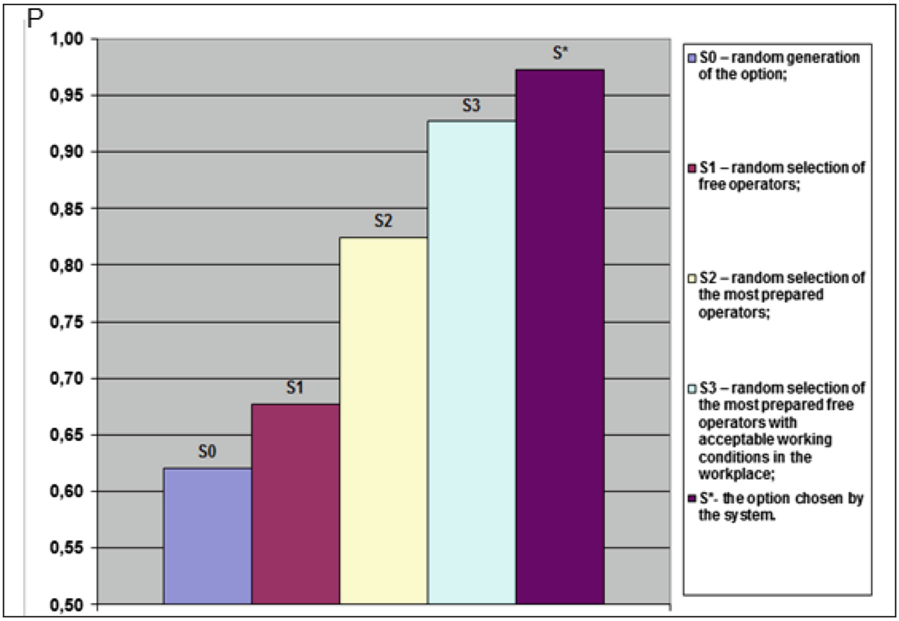


Fig. 3. The average probability of error-free and timely execution for request (P), depending on the chosen strategy (S * - the variant proposed by the decision support system using the whole complex of models).

5 Conclusion

One of the most effective tools to increase the reliability of information systems and technologies is the development and implementation of an ergonomic quality assurance system. Support of ergonomic decision-making can be effective only if using formal models that provide an operative computer evaluation of working conditions of operators, activity algorithms, and the effectiveness of alternative variants of ergonomic measures. The complex of models makes it possible to determine the structure of information support for the management system of ergonomic quality of information processing and control systems.

The obtained results are brought to the level of engineering developments and can be used in the services of ergonomic support of "human-technology-environment" systems with a discrete nature of activity.

The results allow for making decisions on the use of ergonomic reserves to increase the efficiency of human-machine interaction.

References

1. Ge, Y., Wang, A., Cheng, J.: A spatial scheduling strategy for irregular workplace in shipbuilding. In: IEEE International Conference on Mechatronics and Automation, ICMA 2016, pp. 12–16. IEEE (2016). https://doi.org/10.1109/icma.2016.7558526
2. Tanizaki, T., Shimmura, T., Fujii, N.: Shift scheduling to improve customer satisfaction, employee satisfaction and management satisfaction in service workplace where employees and robots collaborate. In: Hara, Y., Karagiannis, D. (eds.) Serviceology for Services, ICServ 2017. Lecture Notes in Computer Science, vol. 10371, pp. 15–25. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61240-9_2
3. Kempa, W.M., Woźniak, M., Nowicki, Robert K., Gabryel, M., Damaševicius, R.: Transient solution for queueing delay distribution in the *GI*/*M*/1/*K*-type mode with "Queued" waking up and balking. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2016. LNCS (LNAI), vol. 9693, pp. 340–351. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39384-1_29
4. Cacciabue, P.C.: Human error risk management for engineering systems: a methodology for design, safety assessment, accident investigation and training. Reliab. Eng. Syst. Saf. **83**(2), 229–240 (2004). https://doi.org/10.1016/j.ress.2003.09.013
5. De Felice, F., Petrillo, A.: Methodological approach for performing human reliability and error analysis in railway transportation system. Int. J. Eng. Technol. **3**(5), 341–353 (2011)
6. De Felice, F., Petrillo, A., Carlomusto, A., Ramondo, A.: Human reliability analysis: a review of the state of the art. Int. J. Res. Manag. Technol. (IJRMT) **2**(1), 36–41 (2012)
7. Dul, J., Bruder, R., Buckle, P., Carayon, P., Falzon, P., Marraset, W.S.: A strategy for human factors/ergonomics: developing the discipline and profession. Ergonomics **55**(4), 377–395 (2012). https://doi.org/10.1080/00140139.2012.661087
8. Boiko, A., Shendryk, V.: System integration and security of information systems. Procedia Comput. Sci. **104**, 35–42 (2017). https://doi.org/10.1016/j.procs.2017.01.053
9. Adamenko, A.N., Asherov, A.T., Lavrov, E.A., et al.: Information controlling human-machine systems: research, design, testing, Reference book. In: Gubinsky, A.I., Evgrafov, V. G. (eds.) Mechanical Engineering, Moscow (1993). (in Russian)
10. Aalipour, M., Ayele, Y.Z., Barabadi, A.: Human reliability assessment (HRA) in maintenance of production process: a case study. Int. J. Syst. Assur. Eng. Manag. **7**(2), 229–238 (2016). https://doi.org/10.1007/s13198-016-0453-z
11. Akyuz, E., Celik, M.: Computer-based human reliability analysis onboard ships. Procedia – Soc. Behav. Sci. **195**, 1823–1832 (2015). https://doi.org/10.1016/j.sbspro.2015.06.398
12. Bertolini, M., Bevilacqua, M.: Fuzzy cognitive maps for human reliability analysis in production systems. In: Kahraman, C., Yavuz, M. (eds.) Production Engineering and Management under Fuzziness. Studies in Fuzziness and Soft Computing, vol. 252, pp. 381–415. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-12052-7_16
13. Martins, M.R., Matuna, M.C.: Application of Bayesian Belief networks to the human reliability analysis of an oil tanker operation focusing on collision accidents. Reliab. Eng. Syst. Saf. **110**, 89–109 (2013). https://doi.org/10.1016/j.ress.2012.09.008
14. Anokhin, A., Ivkin, A.: Evaluation of ecological interface design for supporting cognitive activity of nuclear plant operators. In: Proceedings of the 5th International Conference in Applied Human Factors and Ergonomics 2014 and the Affiliated Conferences, pp. 260–270 (2014)

15. Lavrov, E., Pasko, N., Tolbatov, A., Tolbatov, V.: Ergonomic reserves for improving reliability of data processing in distributed banking systems. In: Proceedings of 2nd International Conference on Advanced Information and Communication Technologies-2017 (AICT-2017), pp. 79–82, Lviv, Ukraine, 4–7 July 2017 (2017)
16. Sedova, N.A., Sedov, V.A., Bazhenov, R.I.: Analysis of emergency level at sea using fuzzy logic approaches. In: Hu, Z., Petoukhov, S., He, M. (eds.) Advances in Artificial Systems for Medicine and Education, AIMEE 2017. Advances in Intelligent Systems and Computing, vol. 658, pp. 314–322. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-67349-3_30
17. Sedov, V.A., Sedova, N.A., Glushkov, S.V.: The fuzzy model of ships collision risk rating in a heavy traffic zone. Vibroengineering PROCEDIA **8**, 453–458 (2016)
18. Rotshtein, A., Shtovba, S.: Modeling of the human operator reliability with the aid of the Sugeno fuzzy knowledge base. Autom. Remote Control **70**(1), 163–169 (2009)
19. Lavrov, E., Pasko, N.: Approach to optimization models bank forming for distributing of functions between the ACS operators. Eastern-European Journal of Enterprise Technologies. Ser. “Mathematics and Cybernetics – Fundamental and Applied Aspects” 1(4/49), pp. 46–50 (2011). (in Russian)
20. Lavrov, E., Pasko, N., Krivodub, A., Tolbatov, A.: Mathematical models for the distribution of functions between the operators of the computer-integrated flexible manufacturing systems. In: Proceedings of the XIII-th international conference TCSET-2016 “Modern Problems of Radio Engineering, Telecommunications, and Computer Science”, Lviv-Slavsko, Ukraine, 23–26 February 2016, pp. 72–76 (2016)
21. Lavrov, E., Pasko, N.: Approach to the formalized description of discrete activity in the systems “man-technology-environment”. Periodical of the Sumy State University. Series Technical Sciences 3, pp. 55–67 (2012)
22. Lavrov, E., Barchenko, N., Pasko, N., Borozenec, I.: Development of models for the formalized description of modular e-learning systems for the problems on providing ergonomic quality of human-computer interaction. Eastern-Eur. J. Enterp. Technol. Ser. Inf. Technol. **2**(2/86), 4–13 (2017). https://doi.org/10.15587/1729-4061.2017.97718
23. Lavrov, E., Barchenko, N., Pasko, N., Tolbatov, A.: Development of adaptation technologies to man-operator in distributed e-learning systems. In: Proceedings of 2nd International Conference on Advanced Information and Communication Technologies-2017 (AICT-2017), Lviv, Ukraine, 4–7 July 2017, pp. 83–87 (2017)
24. Lavrov, E., Pasko, N., Krivodub, A., Barchenko, N., Kontsevich, V.: Ergonomics of IT outsourcing: development of a mathematical model to distribute functions among operators. Eastern Eur. J. Enterp. Technol. **4**(80), 32–40 (2016). https://doi.org/10.15587/1729-4061.2016.66021

Business Intelligence for Information and Software Systems: Special Session on Intelligent Methods for Data Analysis and Computer-Aided Software Engineering

Adaptive Resource Provisioning and Auto-scaling for Cloud Native Software

Olesia Pozdniakova(✉), Dalius Mažeika, and Aurimas Cholomskis

Vilnius Gediminas Technical University, Saulėtekio al. 11, 10223 Vilnius, Lithuania
{olesia.pozdniakova,dalius.mazeika,aurimas.cholomskis}@vgtu.lt

Abstract. Cloud-native applications (CNA) are developed to run on the cloud in a way that enables them to fully exploit the cloud computing characteristics. These applications are strongly dependent on automated machinery (i.e. auto-scaling engines, schedulers and cloud resource provisioning software), which enables elasticity and auto-healing. These features improve application availability, resource utilization efficiency and help minimizing SLA violations related to performance. This work provides a generic architecture of software system that enables elasticity of cloud native software by use of automated scaling and resource provisioning. The architecture is based on analysis of previous works presented by practitioners and academia. Also it is a cloud platform and vendor agnostic.

Keywords: Auto-scaling · Cloud native · Adaptive Resource provisioning · Microservice

1 Introduction

A cloud platform provides virtually unlimited amount of on-demand resources on pay-as-you go basis. Commonly user is charged for cloud resources utilization based on time or capacity, that triggers a need to utilize resources as efficiently as possible. Cloud elasticity allows efficient resource utilization [7]. The elasticity is achieved through automated scaling of cloud resources, e.g. databases, storage, virtual machines (VM's) or/and container instances.

Software developed to run on top of a cloud is commonly referred as cloud native application (CNA). Such applications are developed to be elastic and fully exploit the automation features of cloud computing platform [3,10,17]. Fehling et al. [6] define properties of cloud native application as IDEAL (Isolation, Distribution, Elasticity, Automated management, Loose coupling). Even though isolation, distribution and loose coupling are applicable to classical distributed systems development, elasticity, auto-healing and use of automation are the most significant properties of cloud native application. Cloud native application can utilize all types of cloud service models [12]. It can run on top

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 113–129, 2018.
https://doi.org/10.1007/978-3-319-99972-2_9

of Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and might be composed of Software-as-a-Service (SaaS) components. Different approaches must be applied to enable adaptive auto-scaling. Selection of approach might depend on type of service model used to host cloud native application and on type of application.

This work introduces a high-level generic architecture that consists of a set of functional components that are required to enable auto-scaling of a cloud native applications. The architecture takes into account that auto-scaling should not be limited to virtual machines and containers provisioning, as application might utilize different type of cloud service model (IaaS or PaaS) and can run on different clouds. This document provides examples of implementations that are met in the industry and academia case studies, however most of these studies are concentrating on one or two components. This document shows generic components required to enable auto-scaling of CNA, abstracting from implementation details. Nevertheless, document will provide few examples of possible implementations to illustrate a purpose of components and how they interact with each other. Worth mentioning that proposed solution best suits for applications that are developed using microservice architecture (MSA), however it can be applied to other type of applications running on public or private cloud solutions. It targets large scale applications designed to run on public cloud platforms. The architecture can be implemented using existing products with addition of several custom components which will be discussed later in this document.

This documents is structured as follows: Sect. 1 is an introduction to the topic; Sect. 2 defines the problem background and related works; Sect. 3 starts with an overview of proposed reference architecture and its components that enable adaptive resource provisioning and auto-scaling of cloud native application and ends with more detailed introduction to architecture components implementation approaches, examples and products. Last section concludes this work and gives future work directions.

2 Background and Related Work

It becomes more and more popular to deliver software as a service through the cloud platforms. Software providers aim to deliver such services in a cost effective way and at the same time meet service level agreements, which might include software availability, response times and/or amount of transactions or number of clients served per specific period of time. Auto-scaling is a popular approach to overcome SLA issues related to lack of system resources. However, auto-scaling provided by cloud providers are commonly based on simple reactive rules or time schedules, which are not very efficient for large scale applications. Another thing to consider is that modern SaaS are commonly built using microservice architecture, which brings more granularity into size of applications components. This improves elasticity and portability, however, it brings higher complexity in comparison to traditional N-tier applications. Further, CNA can utilize cloud provider's PaaS services, which provide capacity through various service tiers or subscriptions, so cloud resource auto-scaling problem is not limited only to automated IaaS resource provisioning, but also to PaaS and even other SaaS. As result, enablement of adaptive auto-scaling and resource provisioning for cloud native application that runs on different cloud platforms and across different cloud models (IaaS and PaaS) requires a complex software system that consist of multiple components. In contrast to previous works [2,8,11,19], which are mainly concentrating on auto-scaling algorithms for virtual machines or containers (IaaS), this document provides a conceptual architecture that takes into account the fact that CNA can utilize different cloud service models and can run on different cloud platforms. Kratzke and Quint [10] provide an overview of conceptual architecture for CNA. The solution introduced in this document goes one step deeper by introducing possible implementation approaches and products that can be used for CNA and auto-scaling solutions. The architecture is inspired by self-managing cloud native application architecture proposed Toffetti et al. [19]. However, their work concentrates on auto-healing and self-management functionality, where this document concentrates on auto-scaling across all cloud service models (IaaS and PaaS). The main contributions of this article are:

- a holistic view of architecture of adaptive resource provisioning and auto-scaling system for cloud native software;
- overview of functional components and tools that can be used in real life deployments;
- the architecture serves specifically for auto-scaling needs of CNA irrespectively of what cloud service model (IaaS or PaaS) is used.

Next section defines the proposed architecture and its components that support or enable cloud native application adaptive auto-scaling and resource provisioning.

3 Proposed Architecture

This section introduces a generic architecture that enables adaptive auto-scaling and resource provisioning for CNA. The aim of the proposed architecture is to enable a cloud native application automatic scalability, resiliency and cloud platform independence.

Adaptive cloud native application is a complex software system that consist of multiple components that enable the application's ability to be easy scalable and compliant with SLA requirements. Figure 1 provides a conceptual view of such system architecture. The presented architecture components can be split into two groups. The first group's components enable application scalability (upper grayed area in Fig. 1) and the second ones enable cloud platform and model independence (bottom grayed area in Fig. 1). Toffetti et al. [19] have proposed similar architecture, however it was mainly oriented on IaaS (VM's or containers) and application auto-healing. The architecture in this document concentrates on auto-scaling and takes into account that application can run on top of PaaS, not

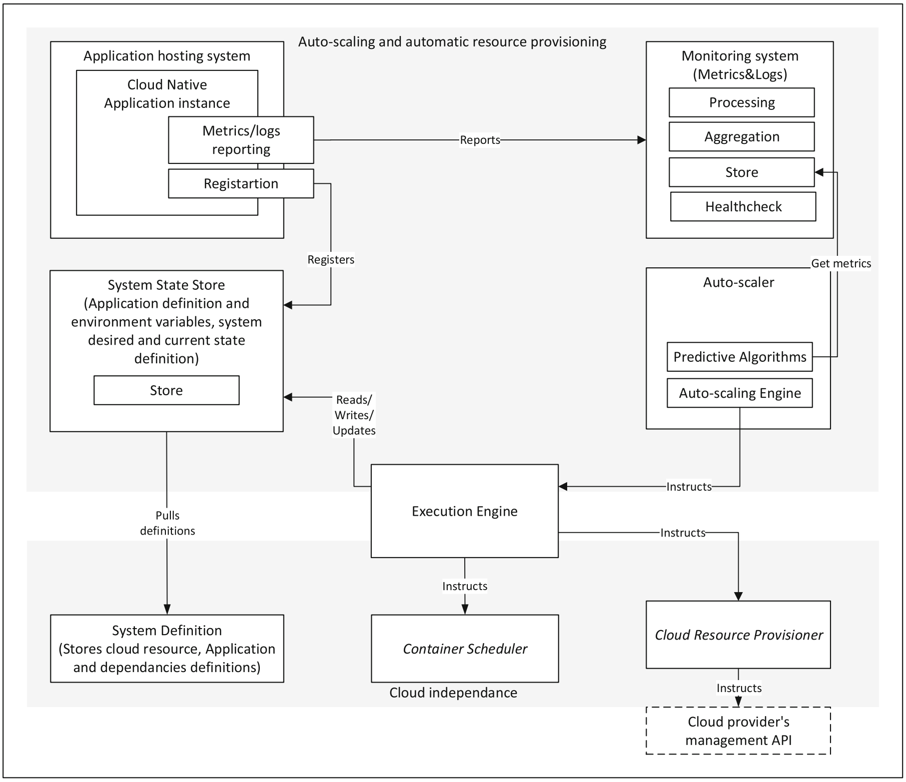


Fig. 1. Principal schema of proposed architecture

only IaaS. It also has cloud provisioner layer, that acts as abstraction layer between cloud native software system and cloud providers management API. This together with execution engine decouples application from cloud platform and removes a need to build custom logic in application itself.

Since CNA interacts with or relies on other components (e.g. schedulers, orchestrator, service discovery, cloud provisioner service) it might seem that all these components are highly coupled and thus must be either developed from scratch together with a CNA or produced by a single vendor. However majority of components are already developed as can be seen from Table 1. Table 1 provides short description of components and products that might be used when designing system.

Table 1. Description of architecture components.

Functional component	Description and implementation
Application instance (Sect. 3.1)	This is a **a CNA application instance** as defined in Sect. 3.1. In order to to support auto-scaling it should be able to report its logs and metrics (input for auto-scaling decisions) CNA should be centrally registered in order to be automatically discoverable. Instance can perform self-registration or be registered by 3rd party components to become discoverable by other services and/or components Reporting and registration can be done using custom code, libraries or frameworks, like Spring Boot or Cloud, Micro, Steeltoe. It also can be done using 3rd party products like containers i.e. cAdvisor, Beats, or monitoring agents, e.g. FluentD, statsD and service discovery agents, like Consul, Eurika, etcd
System state store (Sect. 3.2)	Auto-scaler must be aware about system state, dependencies between components and limitations. This component is used to store running software configuration and state, participate in service registration and discovery process. It keeps configuration parameters and application data like **instance state (booting, running, shutting down, etc.), port, application ID, type (web, data base), workload pattern and other**. This component perform the following functions: – it fetches **application definition** data from system definition component and stores configuration data that later is used by execution engine – it centrally **stores application persistent data**, that can be used for application restoration (auto-healing), conflict and dependency resolution, auto-scaling (see Sect. 3.5 for more details) – **service discovery** (client/service gets information about services from the store) System state store can be implemented using distributed and strongly consistent store. Such solutions commonly include agents, scripts or background services that enable service health-checks/heartbeat functionality, dynamic configuration updates. Examples of stores, that are also used for service discovery are Eurika, Consul, Zookeeper, etcd, DNS and others
Monitoring system (Sect. 3.4)	Auto-scaling engine needs information about resource utilization. Monitoring system is a system that collects, aggregates and stores application and/or system **logs and metrics**. Monitoring system can be implemented as a single solution or as a set of multiple distributed components. The modern tools commonly use time-series or document databases as back-end for storing monitoring and log data. Later, this data is used for SLA monitoring and reporting and auto-scaling. Example of such systems are Elastic (a.k.a ELK) stack, Prometheus, InfluxDB

(*continued*)

Table 1. (*continued*)

Functional component	Description and implementation
Auto-scaling system (Sect. 3.5)	Auto-scaling system makes decisions about need to scale up or down an application and/or its underlaying infrastructure. Most of existing build-in cloud auto-scaling solutions are based on reactive rule-based auto-scaling algorithms. Large scale CNA need to have adaptive auto-scaling engine due to CNA's dynamic nature (see Subsect. 3.5). The adaptive auto-scalers can include components that detect workload pattern or select the most significant metric for auto-scaling decision (see more in Sect. 3.5). As such, it can be composed of several components: – **auto-scaling algorithm** is a core of auto-scaling decision making – **auto-scaling decision making component** informs execution engine (see below) about need for resource provisioning/de-provisioning – **auto-scaling algorithm selection engine** (optional) selects the appropriate auto-scaling algorithm manually or automatically based on monitoring data (see Sect. 3.5)
System definition store (Sect. 3.3)	There should be a definition of nodes, platform and application that is used for new components addition to the system. System definition store component stores CNA infrastructure nodes and application definitions and dependencies between those. These definitions or templates can be reused in any cloud environments. Later, these templates are read by system state store and required application configuration is generated and stored within it. Stored configuration together with system parameters are used by execution engine, once there is a need for auto-scaling action. Mentioned before templates can be implemented using TOSCA standard, or custom resource provisioning and application definition scripts. Use of TOSCA enables solution portability between clouds and dependency management
Execution engine (Sect. 3.6)	Supports auto-scaler with resources provisioning and dependencies tracking. It pulls an application and its underlaying components definitions from system state store and instructs cloud resource provisioner and/or container orchestrator/scheduler about the need to provision resources. It also orchestrates execution and resource provisioning workflow This area was not well developed at the time of this article writing. Cloudify can be an example of similar system, but which does not provide orchestration across public cloud PaaS layer, as such there is a need for custom solution or combination of multiple products

(*continued*)

Table 1. (*continued*)

Functional component	Description and implementation
Cloud resource provisioner (Sect. 3.7)	Component responsible for cloud resource provisioning via cloud providers management API's. It creates an abstraction layer between execution engine and cloud provider – enables cloud platform independence and portability. Popular resource provisioning products are Terraform, OpenStack Heat, CloudFormation, Cloudify
Container scheduler (Sect. 3.8)	Component responsible for container cluster management (auto-healing, start, stop, re-start and similar operations) and containers placement within the cluster (scheduling). It enables cloud platform independence and portability. Example of popular products are Docker Swarm, Kubernets, Nomad, Mesos Marathon

Subsections in this section define the functional components in more details.

3.1 Application Instance

As this document concentrates on resource provisioning and auto-scaling for cloud native applications, it only provides an overview of what cloud native application is. In comparison to not cloud native applications, cloud native applications are designed to be scalable, robust to frequent start, stop and re-start activities and work efficiently within a cloud environment. These applications are designed with use of automation in mind, and are commonly developed using the microservice architecture and container solutions [10] like Docker or rkt, as combinations of those enables more granular application elasticity. Such applications should be stateless or with isolated state – a data that makes application stateful is stored in external storage. This could be either database or caching system. Fehling et al. [6] call this as Isolated state feature of cloud native application.

3.2 System State Store

As CNA consists of multiple components that are provisioned automatically, it becomes hard to track its configuration manually. For that reason, a CNA instance should be able to discover services and be automatically registered in service registry. It should pull its configuration parameters from configuration management store, send logs and metrics to enable auto-healing and auto-scaling features respectfully [19]. System state store components is a part of **service registration and discovery** tools that helps distributed application components to find and talk to each another. Service discovery consists of two major parts: service discovery and service registration.

There are various approaches for service discovery implementation [24]. The most common, that are used for cloud applications, are DNS or key value stores like etcd, Consul, Zookeper and similar.

Service registration can be done either manually or automatically. Automatic registration has two approaches: an application can be developed to register itself or it can use a 3rd party tools like Registrator or Consul agent. Second approach is more flexible in comparison to build into application registration and discovery logic.

System state store enables each new instance of an application or a microservice to store an information about its current state externally. It stores information about service and its definition. This would be data that defines the service instance, i.e. its SLA requirements, status of the service (i.e. active, down, starting), type of service (i.e. stateful, stateless, Web, Application, Database, CPU intensive), context awareness (application knows to which resource or pool it belongs to and amount of running instances), dependencies management (application is aware about components on which it is dependent on and checks on availability of required resources before starting up), configuration variables (IP addresses, ports, environment) and so on. This information is used for this service discovery by other services and by execution engine for controlled auto-scaling logic or for service recovery. For example, after receiving auto-scaling request from auto-scaling engine, execution engine can check amount of a running service instances and compare those to expected amount. If some of instances are still booting up it ignores the request. Another example involves auto-healing component, which logic is similar to one that is described in [19] work. So we won't explicitly describe it here. Auto-healing logic might impact auto-scaling, if auto-healing component initiates removal of failed component and replacement of it with healthy one. A temporary increase in system load can happen at this moment. For that reason, information about auto-healing activity must be stored in system state store. When application is restoring its state, execution engine needs to make sure that no additional instances are provisioned as part of auto-scaling component as reaction to temporary increase in load. This can be done by comparing amount of in provisioning state instances against total amount of required instances.

3.3 System Definition

System definition component is used to describe the structure of the application/service that we want to deploy. Applications, containers, VM's (if needed) and cloud services can be defined and stored in a single place. This helps to ensure correctness of configuration and provides a centralized view of the system. Application and infrastructure components can be defined as templates and/or blueprints using Domain Specific Language (DSL) or notations. The templates can be stored in a central repository for centralized retrieval and version control. Example of implementation is provided in Fig. 2.

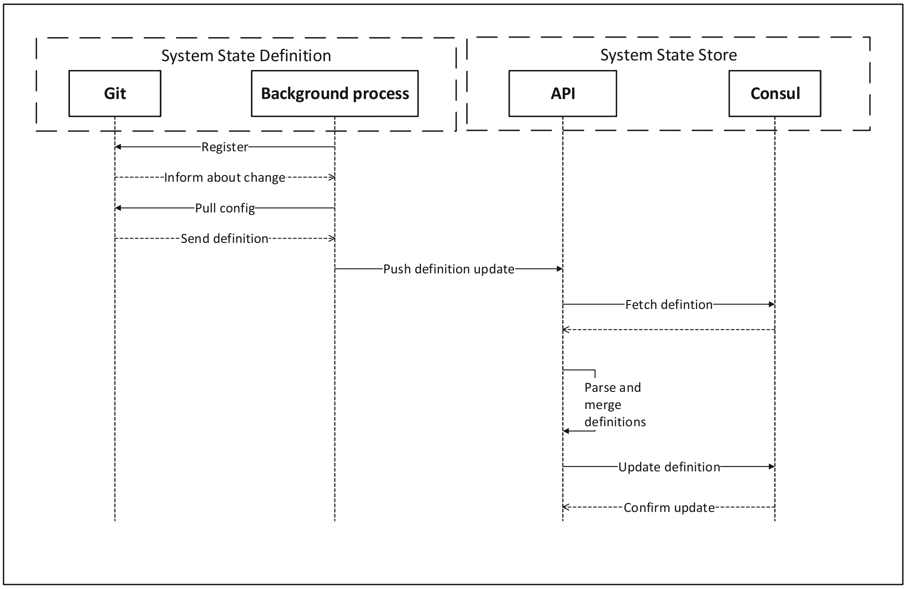


Fig. 2. System definition and system state store interaction example

TOSCA standard can be treated as an example of template definition. It aims to standardize the language for definition of cloud applications and its underlaying infrastructure and platform components. The aim of TOSCA project is to enables cloud platform independent application deployment and setup portability between clouds. It takes a concept of templates that define nodes and applications, and relationships between them. These templates can be understood and processed by tools that are TOSCA-compliant. Cloudify, Apache ARIA, Alien 4 Cloud are example of execution engines that makes these templates actionable (configure and deploy resources as per template definition). Execution engine introduced in this document can act as engine for TOSCA.

3.4 Monitoring System

Microservices and cloud native application auto-scaling strongly rely on monitoring data received from application or infrastructure components, like containers and VM's, hosting the application. For that reason, a set of monitoring tools and approaches, like request identification in tracing [15], were developed for microservices and containers monitoring. Examples of such tools are Prometheus, cAdvisor, Kibana, StatsD, Elastic (a.k.a ELK) stack, Prometheus and InfluxDB and others. More sophisticated list of tools and their overview in cloud applications context can be found in [14, 18].

Monitoring solutions commonly consist of telemetry system that collects application or/and infrastructure metrics. The collected metrics are preprocessed and stored in a database. Later, the metrics can be process by analytics engine and processed data can be used by auto-scaling system. Metrics collected by monitoring system can be split into two categories: system (a.k.a host) metrics and application metrics [9,18].

System metrics are collected from operating system or containers. CPU, RAM, storage I/O utilization and network throughput is an example of host metrics. These are the most popular metrics used for auto-scaling decision making as it is easy to collect and the metrics are independent from application running on a host. Worth mentioning, that set of metrics that can be extracted from containers are limited to metrics that can be exposed by cgroups [1], that overs CPU, RAM, network, storage I/O related metrics, so application metrics need to be collected when more information is required.

Application metrics like response time, request and error rates, workload are examples of application specific metrics. These metrics are usually used for SLA monitoring [2,11,18], rather then for auto-scaling decisions. However, there are several works that propose to use only application metrics for auto-scaling [9,13].

Other important point to consider is usage of aggregated metrics values over a time. The usage of aggregated values rather than instantaneous prevents system from reacting too quickly, or causing rapid fluctuation. This approach also saves database storage and softens impact of cloud resources contentions [5]. Monitoring both types of metrics is required for reliable auto-scaling. For example, application might run on top of PaaS and in this case monitoring data cloud providers platform is required. This data might not include all required information or use too coarse or dense aggregation metrics. Application type (i.e. web, DB) and infrastructure form factor (container, VM) need to be considered when estimating aggregation interval length. These could be retrieved from system state store.

3.5 Auto-Scaling

Auto-scaling is a process of dynamic resource allocation in such way that amount of allocated resources matches performance requirements. Auto-scaling is usually used to achieve the following goals [2,11]:

- solve resource planing problem – automated resource provisioning and application scaling must be done on time to avoid SLA violations;
- solve resources utilization optimization problem – most of time the difference between provisioned and consumed resources should be as low as it is possible.

Several articles provide an overview of auto-scaling techniques for cloud resources that can be grouped into rule-based, control theory based, queuing theory based and machine learning based approaches [2,11,20].

Rule-based auto-scaling is most commonly met approach in the cloud as it is delivered by cloud providers as build-in service. These methods are considered as reactive methods.

Auto-scaling based on control theory, queuing theory and machine learning are considered as proactive approaches. The most of the works related to cloud resource provisioning and auto-scaling uses machine learning algorithms [2,11,20]. The algorithms use resource utilization (host specific parameters) or response time, queue size or amount of sessions (application specific parameters) for auto-scaling decisions. Auto-scaling algorithms that use host/container metrics are easier to deploy and are independent from application, however these might be less accurate in comparison to algorithms based on application specific metrics [11] as well they might have limitations when auto-scaling is done for PaaS platform.

Nevertheless there is a big variety of auto-scaling methods, there is no single algorithm that fits all types of loads [2]. Experiments results presented in [4,13,20,22] show that different algorithms performs differently depending on type of application workload. Manual selection of different algorithms for specific services does not seem optimal due to dynamic nature of CNA. There is a need for adaptive auto-scaling engines. Different approaches are proposed by academia for enabling adaptive auto-scaling:

1. Usage of metrics analytic engine, which would analyze the received metrics and select the most significant one. This could be achieved by applying statistical feature selection techniques like Mutual Information (MI), Information Gain (IG) and Chi-Square Test [4].
2. Usage of pattern analyzing engine, which would identify the type of workload – dynamic (i.e. batch processing) or static. [22] proposes a "black box" auto-scaling, based on machine learning algorithm. [23] uses workload pattern detection mechanism to improve rule-based workload prediction approaches.
3. Usage of auto-scaling algorithm selection engine, which would select best suited auto-scaling algorithm based on type of workload and/or most significant metrics. The efficiency of such approach is evaluated by [13].

Use of this approaches enables adaptive auto-scaling. Based on estimates provided by algorithm, auto-scaling engine would make an auto-scaling decision and inform execution engine about the need to scale up or down.

3.6 The Execution Engine

CNA can be deployed on PaaS or IaaS. In case of PaaS cloud platform provider is responsible for resource auto-scaling, however resources are limited by service plans or tiers. In cases when additional resources are required, migration to new tier or new PaaS service instance is needed. Application owner can control additional resource provisioning and migration between service tiers by use of custom auto-scaling component. The same component can be used for horizontal scaling between service tiers. As execution of such logic would overload auto-scaling component, an execution engine is introduced as separate component. This component can be either centralized (orchestrate the solution) or distributed (management logic is shared between multiple execution agents).

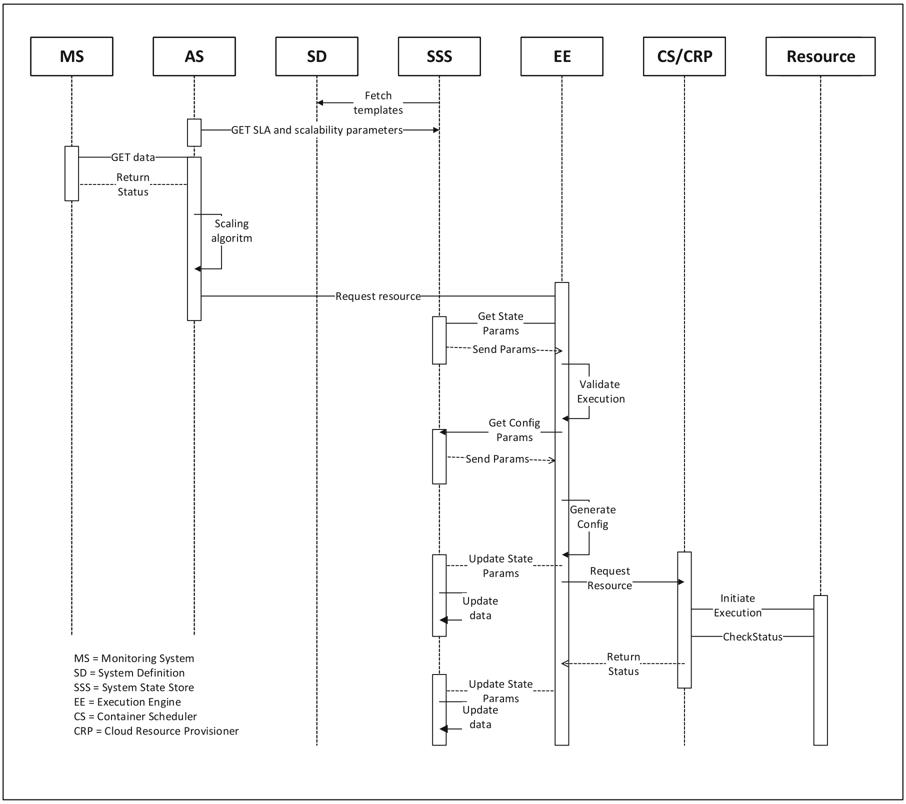


Fig. 3. Auto-scaling execution example

In addition, the execution engine is responsible for application deployment workflow as per service and dependencies definitions stored in a system state and system definition stores. The execution engine communicates with all components to enable provisioning of resources requested by auto-scaler in structured way. It controls resource provisioning requests triggered by administrator, auto-healing and auto-scaling actions in order to avoid race conditions and resource double provisioning. It tracks what resource (container or application instance, VM or vertical scaling of PaaS services) need to be provisioned.

Products available on market cover either one or two of these areas (either infrastructure, platform or application) and, in most cases, are cloud vendor specific. An example of infrastructure oriented systems are Amazon Web Services (AWS) CloudFormation, OpenStack Heat. Cloudify is IaaS and PaaS oriented solution. Example of similar solution is provided in [19]

The sequence diagram shown in Fig. 3 illustrates an example how execution engine enables automated resource provisioning for CNA.

3.7 Cloud Resource Provisioner

Execution engine (EE) can communicate with cloud providers API directly, however this requires development of custom logic for each cloud API. To avoid this EE can use cloud resource provisioners, that are available on a market. It would act as layer of abstraction between execution engine and cloud providers API. As it is mainly used to minimize vendor lock-in and increase solution portability Cloud Resource Provisioner component is marked as an optional component in this architecture.

3.8 Container Scheduler

Container schedulers (a.k.a. container cluster management and scheduler, container orchestrator) are used for optimized container placement on virtual or physical machines. Its tasks also include distribution of workload between cluster machines in case of hosting machine failure. Usage of widely adapted container schedulers minimizes cloud vendor lock. These solutions can be provided as part of PaaS service or deployed on IaaS.

Different types of schedulers were created for cloud computing and later adapted for container scheduling [16,21]. These can be grouped into schedulers used for long living tasks like services, and short living tasks, like AWS lambda or Azure functions. As per [16] schedulers falls into four major types: monolithic, statically partitioned (i.e. Hadoop), two-level (i.e. Mesos), optimistic shared state (i.e. Kubernets, Nomad).

From execution engine point of view, container scheduler is just another component that is used for resource provisioning. However, there might be issues, when auto-scaling component and scheduler are working independently. Usage of monolithic or two-level scheduler might cause some of virtual machines being underutilized, thus auto-scaler might not provide a virtual machine on time. To minimize impact mitigation actions can be performed. For example, set of hosts can be dedicated to a specific type of containers. This potentially minimizes optimal utilization of the hosts as in two-level schedulers case, but at the same time provides environment with similar behavior and utilization patterns, what suitable for predictions. Schedulers based on optimistic shared state algorithm is more preferred as these pride more equal load distribution and perform more efficiently independently from longevity of task [16]. Container schedulers are marked as optional components in this architecture, as cloud native application can have components that are running on top of IaaS, PaaS or "serverless" components. In this case Cloud Resource Provisioner can be used to request mentioned above resources.

4 Conclusions and Future Work

Document provided an overview of architecture and its components required to enable automated resource provisioning of cloud native application. It also introduced products and approaches that can be used to implement these components. Vast majority of these components (System state store, System definition, Monitoring, Cloud resource provisioner, Container scheduler presented in Fig. 1) can be implemented using software components presented on software market (see Table 1 for details). There are many products that can provide required functionality for the proposed system, however there are still gaps in the following areas:

- none of the solution can provide end-to-end resource provisioning across cloud service models (IaaS and PaaS) when system need to be expanded, as solutions are more concentrating on one area either on provisioning server or containers or applications, but not on a full stack;
- auto-scaling solutions provided by cloud service providers are usually based on reactive rule-base algorithms that are static in nature; these limit elasticity of modern dynamic cloud native applications.

The proposed architecture suggests usage of optional mediation layers (container schedulers and cloud resource provisioner's) that enable automated resource provisioning independently from the cloud platform. This can be considered as advantage (brings layer of abstraction and loose coupling between application and cloud platform) and disadvantage (brings extra complexity and additional point of failure).

Execution engine which controls a whole system in a holistic manner (across application, platform and infrastructure layers) has limited implementations (the closest ones are Cloudify and Toffetti [19]). This is not enough to avoid vendor lock-in. Application auto-scaling components (based on reactive algorithms or schedule) are usually provided by cloud vendors. More advanced logic is commonly a responsibility of software engineer. Thus auto-scaler and execution engine components are objects of further research. Validity of this architecture will be tested by practical implementations in the future.

References

1. Runtime metrics — Docker Documentation. https://docs.docker.com/config/containers/runmetrics/
2. Amiri, M., Mohammad-Khanli, L.: Survey on prediction models of applications for resources provisioning in cloud. J. Netw. Comput. Appl. **82**, 93–113 (2017). https://doi.org/10.1016/j.jnca.2017.01.016
3. Andrikopoulos, V., Strauch, S., Fehling, C., Leymann, F.: CAP-oriented design for cloud-native applications. In: Ivanov, I.I., van Sinderen, M., Leymann, F., Shan, T. (eds.) CLOSER 2012. CCIS, vol. 367, pp. 215–229. Springer, Cham (2013). https://doi.org/10.1007/978-3-319-04519-1_14
4. Balaji, M., Aswani Kumar, C., Rao, G.S.V.: Predictive Cloud resource management framework for enterprise workloads. J. King Saud Univ. Comput. Inf. Sci. (2016). https://doi.org/10.1016/J.JKSUCI.2016.10.005. https://www.sciencedirect.com/science/article/pii/S1319157816300921
5. Chen, T., Bahsoon, R.: Self-adaptive trade-off decision making for autoscaling cloud-based services. IEEE Trans. Serv. Comput. **10**(4), 618–632 (2017). https://doi.org/10.1109/TSC.2015.2499770
6. Fehling, C., Leymann, F., Retter, R., Schupeck, W., Arbitter, P.: Cloud Computing Patterns. Springer, Vienna (2014). https://doi.org/10.1007/978-3-7091-1568-8
7. Herbst, N.R., Kounev, S., Reussner, R.: Elasticity in cloud computing : what it is, and what it is not. In: Presented as Part of the 10th International Conference on Autonomic Computing, pp. 23–27. USENIX, San Jose, CA (2013). http://sdqweb.ipd.kit.edu/publications/pdfs/HeKoRe2013-ICAC-Elasticity.pdf
8. Khazaei, H., Ravichandiran, R., Park, B., Bannazadeh, H., Tizghadam, A., Leon-Garcia, A.: Elascale: autoscaling and monitoring as a service. In: CASCON (2017). http://arxiv.org/abs/1711.03204
9. Koperek, P., Funika, W.: Dynamic business metrics-driven resource provisioning in cloud environments. In: Wyrzykowski, R., Dongarra, J., Karczewski, K., Waśniewski, J. (eds.) PPAM 2011. LNCS, vol. 7204, pp. 171–180. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31500-8_18
10. Kratzke, N., Quint, P.C.: Understanding cloud-native applications after 10 years of cloud computing - a systematic mapping study. J. Syst. Softw. **126**, 1–16 (2017). https://doi.org/10.1016/j.jss.2017.01.001

11. Lorido-Botran, T., Miguel-Alonso, J., Lozano, J.A.: A review of auto-scaling techniques for elastic applications in cloud environments. J. Grid Comput. **12**(4), 559–592 (2014). https://doi.org/10.1007/s10723-014-9314-7
12. Mell, P., Grance, T.: The NIST definition of cloud computing. NIST Special Publication 145, p. 7 (2011). https://doi.org/10.1136/emj.2010.096966. http://www.mendeley.com/research/the-nist-definition-about-cloud-computing/
13. Nikravesh, A.Y., Ajila, S.A., Lung, C.H.: An autonomic prediction suite for cloud resource provisioning. J. Cloud Comput. **6**(1), 3 (2017). https://doi.org/10.1186/s13677-017-0073-4. http://journalofcloudcomputing.springeropen.com/articles/10.1186/s13677-017-0073-4
14. Peinl, R., Holzschuher, F., Pfitzer, F.: Docker cluster management for the cloud - survey results and own solution. J. Grid Comput. **14**(2), 265–282 (2016). https://doi.org/10.1007/s10723-016-9366-y
15. Richardson, C.: Microservice Patterns. Manning Pubns Co., Greenwich (2018). https://www.manning.com/books/microservices-patterns
16. Schwarzkopf, M., Konwinski, A.: Omega: exible, scalable schedulers for large compute clusters. In: EuroSys 2013, Proceedings of the 8th ACM European Conference on Computer Systems, pp. 351–364 (2013). https://doi.org/10.1145/2465351.2465386. https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/41684.pdf. http://dl.acm.org/citation.cfm?id=2465386
17. Sethi, M.: Cloud Native Python. Packt Publishing, Birmingham (2017). https://www.packtpub.com/application-development/cloud-native-python
18. Taherizadeh, S., Jones, A.C., Taylor, I., Zhao, Z., Stankovski, V.: Monitoring self-adaptive applications within edge computing frameworks: a state-of-the-art review. J. Syst. Softw. **136**, 19–38 (2018). https://doi.org/10.1016/J.JSS.2017.10.033. https://www.sciencedirect.com/science/article/pii/S016412121730256X
19. Toffetti, G., Brunner, S., Blöchlinger, M., Spillner, J., Bohnert, T.M.: Self-managing cloud-native applications: design, implementation, and experience. Future Gener. Comput. Syst. **75**, 165–179 (2016). https://doi.org/10.1016/j.future.2016.09.002. http://www.sciencedirect.com/science/article/pii/S0167739X16302977
20. Vazquez, C., Krishnan, R., John, E.: Time series forecasting of cloud data center workloads for dynamic resource provisioning. J. Wirel. Mob. Netw. Ubiquit. Comput. Dependable Appl. (JoWUA) **6**(3), 36–53 (2015). https://doi.org/10.22667/JOWUA.2015.09.31.087. http://isyou.info/jowua/papers/jowua-v6n3-5.pdf
21. Verma, A., Pedrosa, L., Korupolu, M., Oppenheimer, D., Tune, E., Wilkes, J.: Large-scale cluster management at Google with Borg. In: Proceedings of the Tenth European Conference on Computer Systems - EuroSys 2015, pp. 1–17 (2015). https://doi.org/10.1145/2741948.2741964. https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/43438.pdf. http://dl.acm.org/citation.cfm?doid=2741948.2741964
22. Wajahat, M., Gandhi, A., Karve, A., Kochut, A.: Using machine learning for black-box autoscaling. In: 2016 7th International Green and Sustainable Computing Conference. In: IGSC 2016 (2017). https://www3.cs.stonybrook.edu/~anshul/igsc16_mlscale.pdf

23. Zhang, L., Zhang, Y., Jamshidi, P., Xu, L., Pahl, C.: Workload patterns for quality-driven dynamic cloud service configuration and auto-scaling. In: Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing, UCC 2014, pp. 156–165 (2014). https://doi.org/10.1109/UCC.2014.24
24. Zhu, F., Mutka, M., Ni, L.: Classification of Service discovery in pervasive computing environments. In: IEEE Pervasive Computing, vol. 4, pp. 81–90 (2005). https://doi.org/citeulike-article-id:1444428. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.90.6859

Cloud Software Performance Metrics Collection and Aggregation for Auto-Scaling Module

Aurimas Cholomskis(✉), Olesia Pozdniakova, and Dalius Mažeika

Vilniaus Gedimino Technikos Universitetas, Vilnius, Lithuania
aurimas.cholomskis@vgtu.lt

Abstract. Cloud computing made a big impact on software architecture evolution. The demand to serve multiple tenants, to include continuous delivery practice into the development process as well as increased system load influenced the style of cloud based software architecture. Microservice architecture is preferred architecture despite its complexity when scalability is an essential attribute of quality of service. Microservices should be managed, i.e., hardware resources should be adjusted based on application load, as well as resiliency should be ensured. Popular IaaS and PaaS providers such as Amazon, Azure or OpenStack ensure auto-scaling and elasticity at the infrastructure level. This approach has the following limitations: (1) Scaling and resiliency is a part of the infrastructure and not emerging from application nature; (2) The software is locked in with a specific vendor; (3) It might be difficult to run and ensure smooth scalability by running software on different vendors at the same time. We are creating auto-scaling module for microservice-based applications. Collecting metrics both at infrastructure and application levels is one important task for auto-scaling. We've chosen to investigate ELK stack and build appropriate architecture around it.

Keywords: Metrics · Autoscaling · Microservice · ELK · Beats Containers · Docker

1 Introduction

Cloud Computing is changing the way of development and deployment of application because of its unique requirements such as higher rate of scalability, reliability, continuous deployment and Integration, and minimum downtime etc. Companies require their enterprise applications to scale on demand, which can be achieved through different cloud computing models. Companies face different challenges while deploying their applications on cloud due to their unique requirements, but the major issue is to scale different functionalities of the applications according to their requirements [1].

Microservice architecture together with elastic infrastructure is a common way to build cloud based applications and meet high scalability requirements. Infrastructure providers usually provide needed tools and infrastructure to ensure elasticity of resources and application.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 130–138, 2018.
https://doi.org/10.1007/978-3-319-99972-2_10

Microservice architecture is a very handy architectural style to implement requirements mentioned above. Each service could be independently deployed and communication with other services established over the network. Such a single service could be scaled out independently of other microservices within application. IAAS or PAAS providers offer ability to scale their virtual machines based on infrastructure metrics, such as CPU, memory, network, etc. These providers offer specials optimized services for microservices based on Docker such as Elastic Container Services from Amazon, or Azure Container Services. However, we claim that it is crucial to monitor application not only at infrastructure level, but also at application level and/or microservice level. Such parameters as response times, amounts of served successful/failed requests are also important for implementing better scaling algorithms. Besides, collecting and aggregating all these parameters together with number of business transactions could be valuable information to identify bottlenecks and ensure service level agreement (SLA) of the cloud application. Moreover, using infrastructure provider services for management functionalities results in vendor lock-in effectively preventing cloud applications to adapt and run on the most effective cloud for the job [2].

Self-managed or self-adaptive microservices is a novel concept to manage auto-scaling and healing of microservices at application level. It allows to build elastic application independently on infrastructure provider and enables particular application services to run on the most effective cloud for the job.

Auto-scaling of the application is heavily dependent on application and infrastructure performance metrics. Both reactive and predictive auto-scaling algorithms need most recent and historical data for larger interval of time, e.g. CPU load, memory usage, response times, request count and others. In microservice architecture it is even a more challenging task, since each individual microservice should be monitored, while a single application could contain dozens of different microservices.

Our ambition is to create self-adaptive microservice architecture which able to monitor, scale out or downscale itself independently on any infrastructure provider. One of the task is to establish an architecture with appropriate tools which is able to collect metrics from infrastructure providers, containers and microservices. These data should be stored in efficient data storage and able to serve appropriate metrics fast enough for auto-scaling engine. Based on data auto-scaling engine could execute reactive and predictive scaling algorithms and send instructions to any infrastructure provider to execute scaling.

Few metrics collection potential issues are addressed in this paper. (a) Infrastructure measurements are usually provided by popular IAAS or PAAS providers, however, for our task we need to collect application level metrics as well as it should be as much real time as possible. (b) The application is always dynamic - its services scales or downscales all the time. This mean that together with metrics we should collect and store application nodes topology. (c) The data amounts would be huge and storage issues should be addressed. Data querying should be measured as it has impact of our auto-scaling module reaction times. (d) Last but not least, the questions how much data we need to collect that data amounts should be feasible to store and execute quick enough queries for our task. Also it is not clear yet how many attributes should be stored and how much valuable they are for auto-scaling module.

2 Related Work

In order to solve our issues addressed we reviewed articles related with cloud based application metrics collection and those which has a similar goals to establish the auto-scaling mechanism for the application. The most similar work we found in the Hamzeh Khazaei *et al*. article where ELK stack (Elasticsearch, Logstash, Kibana) is used for metrics collection and auto-scaling [3]. This study was an inspiration to focus on ELK stack as it showed it might be a proven solution for an auto-scaling microservice based application together with Docker containers. Authors has the whole idea to have scaling as a service -metrics collection module together with auto-scaling module are not incorporated in the specific application architecture. However, reviewed article lacks of details about metrics collection and aggregation. Furthermore it collects container level metrics only, where our goal is to measure performance at application level as well. We think, metrics collection could have challenging issues, because of multiple data sources and huge data amounts, e.g. application of dozens microservices running could produce megabytes of data every second. Also authors are using only straightforward reactive scaling algorithms which requires tuning for each application, where our goal is to apply machine learning algorithms for predicting scaling. Thus much more metrics data is needed to store, fetch and analyze.

We also reviewed articles regarding Elasticsearch performance. Liberios Vokorokas *et al*. has done a research of non-relational databases [4]. They compared MongoDB with Elasticsearch performance in terms of read operations. Elasticsearch performance benchmark indicated very good results i.e. 900 of 3500 documents were able to read in little more than 50 ms. Sheffi Gupta also performed a research of Elasticsearch performance [5]. He benchmarked read operations with 20 000 documents and on average it took more than 80 ms to query such amount of data. In both cases the results were very good and it seems that it has a linear time dependency over fetched number of documents. On the other hand we estimate to have above 20 000 documents which are needed to fetch over the short period of time, while millions of documents are stored. So our goal is to make benchmarking with much larger amount of data and to analyze scalability.

Some previous works [6–9] using a machine learning algorithms to fulfill their purposes and it worthwhile to evaluate those articles for auto-scaling module. Commonly, all these articles not providing details about metrics collection and storage. This might be related due to fairly simple metrics collection and it not worth for authors to provide more details. Some of these articles describe collecting metrics from infrastructure providers and these are heavily tied with infrastructure provider tools, such as CloudWatch from AWS. Our goal is an opposite - to be IAAS or PAAS providers' agnostic and collect metrics also from application. This requires more extensive research in metrics collection.

3 Design and Implementation

3.1 Background

Docker

Docker container is an emerging container technology which provides different functionalities to make provisioning of microservices cost effective and easy to manage [10]. Docker containers implementing immutable server concept, where with an immutable server, you make each change to a base image, and then you know that all instances created from that image are consistent. The main differences between instances of a server role come from configuration settings, which should come from outside the server [11].

In our design, metrics collection is based on microservices deployed as Docker containers. It is for easier configuration and deployments, but it is not a limitation for the tasks we are solving. Metrics could be collected not only from Docker engine, but also from regular virtual machines. Other supporting services such as service registry, ELK or beats we are deploying as Docker containers due to faster configuration and faster development turnaround.

ELK

Elasticsearch, Logstash and Kibana are the open source tools. Elasticsearch is a distributed JSON-based search and analytics engine, Logstash is server side data processing pipeline that ingests data from multiple sources simultaneously, transforms it and then sends it to Elasticsearch. Kibana visualizes data with charts and graphs in Elasticsearch [12].

ELK is very popular tool for application logs aggregation and visualizing those. However it is getting more and more popular to store metrics as well. In our architectural approach ELK stack plays a critical role, as we need to ensure that metrics are shipped fast enough, although the data querying would take a reasonable amount of time. As well as think how data from different sources should be stored, e.g. we need to have our instance topology at a given time.

Beats

The Beats are open source data shippers that you install as agents on your servers to send different types of operational data to Elasticsearch. Beats can send data directly to Elasticsearch or send it to Elasticsearch via Logstash, which you can use to parse and transform the data [13]. Currently, there exist more than 60 different Beats, developed by Elastic and the community in the Beats family. We will use several beats in our implementation to ship different metrics to ELK.

3.2 Proposed High Level Architecture

To deal with mentioned issues earlier the proposed architecture based on ELK stack. Standard ELK stack is well known solution for logs aggregation mainly. Together with the Beats we extended the standard solution to improve cloud based applications monitoring, in order to be able to scale efficiently. Application itself should not be dependent on metric collection or auto scaling logic. This means that auto-scaling application must be pluggable as much as possible with minimal configuration adjustment effort. Application architecture itself should not be relevant – it could have dozens of microservices, or be a single service monolith (Fig. 1).

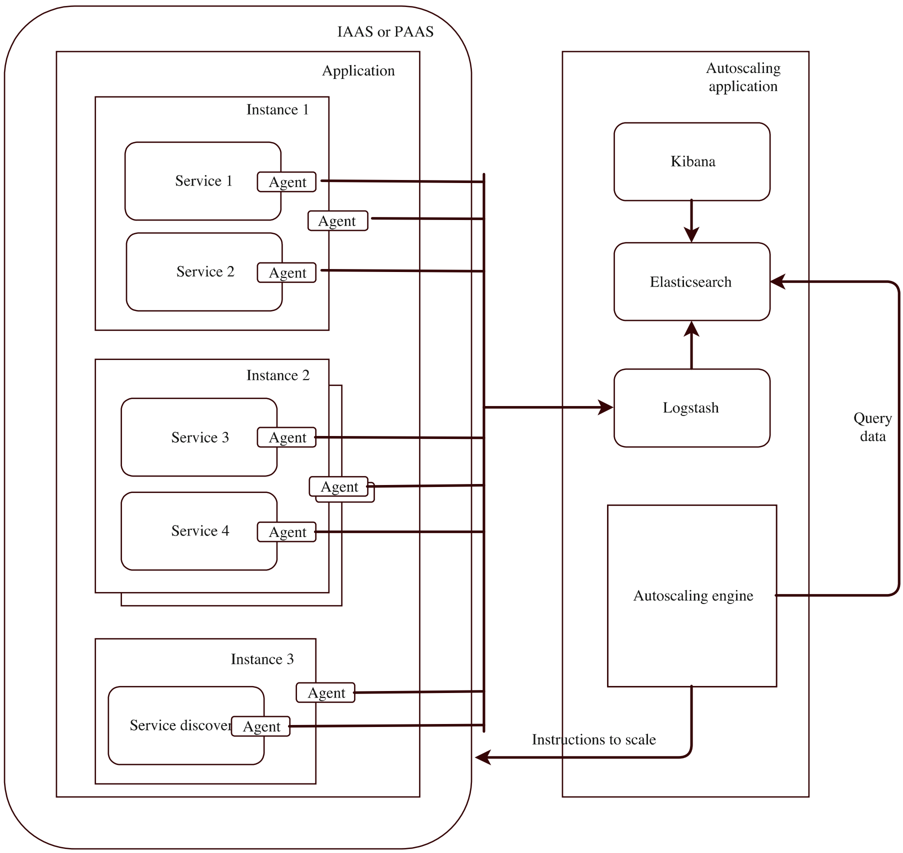


Fig. 1. Principal schema of application metrics collection

Beat agents must be deployed next to the application. There are no limitations or restrictions about the Beats. It could be Dockbeat for Docker, Springbeat for JAVA based Spring applications, or it could be Cloudwatchmetricbeat which is able to read metrics from AWS Cloud Watch service. Moreover it could be used universal beats, like Httpbeat which calls defined APIs and push results to ELK. With beats we're able to collect all kind of metrics at any level, starting with infrastructure level and ending in collecting application specific measurements.

Collected metrics are pushed to the Logstash where data filtering, transformations or tagging are executed and pushed forward to Elasticsearch for indexing and storing. Auto-scaling engine queries data from Elasticsearch, applies needed algorithms and sends instructions to infrastructure provider or Docker engine to scale out or downscale particular services (Fig. 2). At this point auto scaling engine is represented as a single entity, but in later research stages this could be much more complex component.

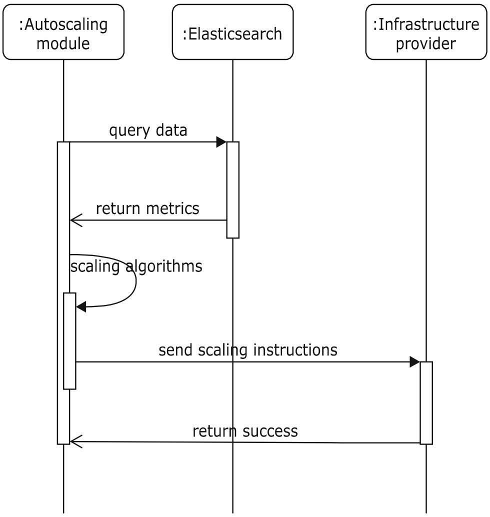


Fig. 2. Auto-scaling engine flow

To establish a proof of concept of metrics collection, we used a Spring Cloud based application deployed on AWS EC2 instances within Docker containers. Spring Cloud provides tools for developers to quickly build common patterns in distributed, microservice based, systems. In our case service discovery is used also. Service discovery is very important part of microservice architecture – all microservices register to service discovery and it distributes others location to every microservice. Besides that, it could ensure health check and store configuration based on key value store. We chose Consul as service discovery. Service discovery plays very important role in auto-scaling engine – it should we aware how many instances are up at a given moment. In other words, we need to have an instance topology together with other metrics.

Agents are deployed also as separate containers. We are using Metricbeat, Dockbeat, Springbeat and ConsulBeat to ship various metrics to ELK.

4 Validation and Results

We used ELK deployed as Docker container on Amazon EC2 t2.medium instance for benchmarking. It contains 2 CPU and 4 GB of RAM. It is a minimal recommended setup for running ELK stack. We emulated test data of 4 000 000 metric documents. A single document represents a single metric, e.g. memory usage with a timestamp. It also contains additional metadata, such as container name, memory available, etc. It is schema-less JSON format – documents with a different format are produced for different type of metrics.

With our first benchmarks we need to check how fast we could query documents depending on their amount. We measured two factors – query time which is provided by Elasticsearch and whole request time which we are measuring. As shown in Fig. 3, there are obvious linear dependency between data amount and request execution time. Request durations are very acceptable since our goal is to execute scaling logic in no more than 30 s including, querying data, executing calculation algorithms and send instructions to infrastructure providers. Blue line is Elasticsearch query time, where red one is request time. It must be noticed that large number of documents produces quite large amount of data where network might be a bottleneck.

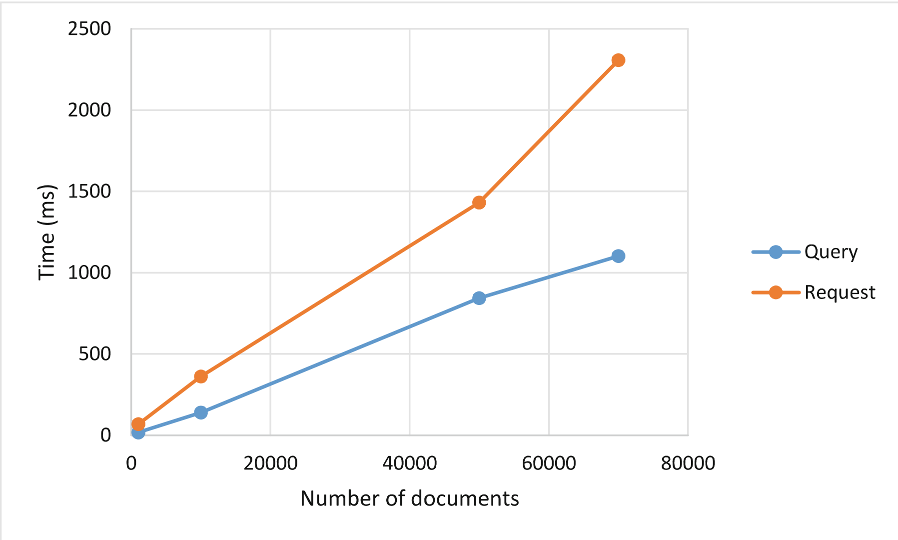


Fig. 3. Query execution times

Initially we had no clue how long single document processing takes time. We made an assumption that increasing number of agent could have an impact of documents processing time. Beats sends documents to Logstash for processing first and then those documents are forwarded for indexing to Elasticsearch. Logstash ensures controlled flow of documents – it means that if it is not able to process data shipped it informs Beats to slowdown shipping. That behavior is sufficient for log aggregation, but for metrics we need to have metrics data as soon as possible. After performing benchmarks we found out that processing time is pretty stable despite the number of Beats. We noticed that processing time correlated with document size. However benchmarking 50-100 agents ends up as expected, that Logstash was not capable to handle the load, so the delays were more than a minute. This could be easily solved by scaling out Logstash instances, but for our benchmarking purposes and simplicity it is enough of 20 Beats agents to be capable to handle (Fig. 4).

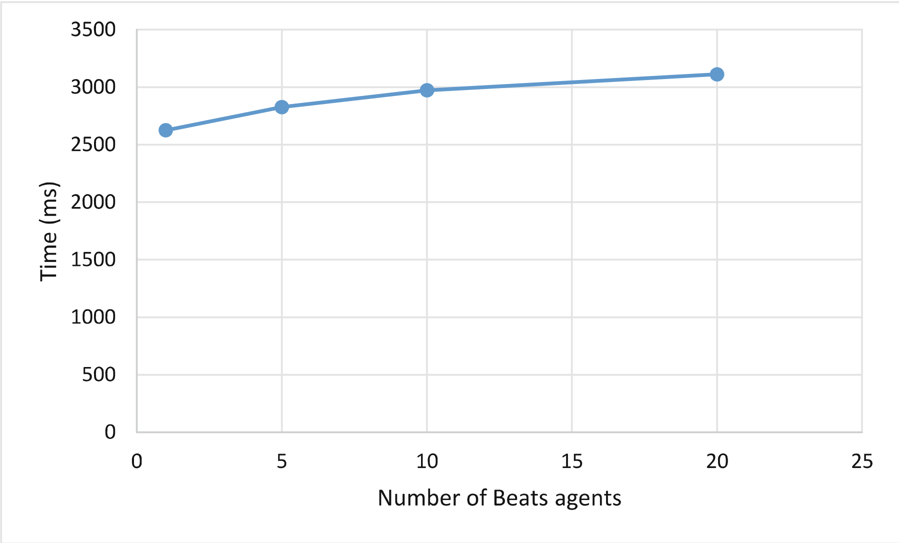


Fig. 4. Document processing time

Together with infrastructure metrics we were able to store application specific metrics, e.g. data from Spring Boot actuator as well as data from Consul registry service. All these data stored indexed and stored as a time series. This allows quickly access all the data and understand how system was performing at any given time.

5 Conclusions

In this paper we evaluated ELK stack together with Beats as a toolset for metric collection and aggregation. This component is crucial for auto-scaling engine, which needs to fetch correct data fast for reactive and predictive algorithms. We were able to validate that Elasticsearch is able to cope with large amounts of data, execute queries and provide results extremely fast. Although we benchmarked that setup is able to cope with 20 agents without significant degradation of shipped documents processing time. Furthermore, despite infrastructure metrics collection we were able to collect application service list together with health status from Consul and also some application specific metrics. There is no need to worry about data amounts collected from Elasticsearch perspective, since it executes queries extremely fast. However, some optimizations for data could be made to avoid high traffic over the network.

References

1. Singh, V., Peddoju, S.K.: Container-based microservice architecture for cloud applications. In: International Conference on Computing, Communication and Automation (ICCCA 2017), pp. 847–852 (2017)
2. Toffetti, G., Brunner, S., Blochlinger, M., Dudouet, F., Edmonds, A.: An architecture for self-managing microservices. In: AIMC 15 Proceedings of the 1st International Workshop on Automated Incident Management in Cloud (2015)
3. Khazaei, H., Ravichandiran, R., Park, B., Bannazadeh, H., Tizghadam, A.: Elascale: Autoscaling and Monitoring as a Service. In: CASCON 2017 (2017)
4. Vokorokos, L., Uchnár, M., Leščišin, Ľ.: Performance optimization of applications based on non-relational databases. In: 2016 International Conference on Emerging eLearning Technologies and Applications (ICETA) (2016)
5. Gupta, S., Rani, R.: A comparative study of elasticsearch and CouchDB document oriented databases. In: International Conference on Inventive Computation Technologies (ICICT) (2016)
6. Alipour, H., Liu, Y.: Online machine learning for cloud resource provisioning of microservice backend systems. In: 2017 IEEE International Conference on Big Data (Big Data), pp. 2433–2441 (2017)
7. Iqbal, W., Dailey, M.N., Carrera, D.: Unsupervised learning of dynamic resource provisioning policies for cloud-hosted multitier web applications. IEEE Syst. J. **10**(4), 1435–1446 (2016)
8. Sun, D., Fu, M., Zhu, L., Li, G., Lu, Q.: Non-intrusive anomaly detection with streaming performance metrics and logs for DevOps in public clouds: a case study in AWS. IEEE Trans. Emerg. Topics Comput. **4**(2), 278–289 (2016)
9. Jindal, A., Podolskiy, V., Gerndt, M.: Multilayered cloud applications autoscaling performance estimation. In: 2017 IEEE 7th International Symposium on Cloud and Service Computing (SC2), pp. 24–31 (2017)
10. Singh, V., Peddoju, S.K.: Container-based microservice architecture for cloud applications. In: 2017 International Conference on Computing, Communication and Automation (ICCCA), pp 847–852 (2017)
11. Kief Morris: ImmutableServer. https://martinfowler.com/bliki/ImmutableServer.html. Accessed 26 Mar 2018
12. ELK Stack. https://www.elastic.co/elk-stack. Accessed 26 Mar 2018
13. Beats reference. https://www.elastic.co/guide/en/beats/libbeat/current/beats-reference.html. Accessed 26 Mar 2018

Application of SEO Metrics to Determine the Quality of Wikipedia Articles and Their Sources

Włodzimierz Lewoniewski[1(✉)], Ralf-Christian Härting[2], Krzysztof Węcel[1], Christopher Reichstein[2], and Witold Abramowicz[1]

[1] Poznan University of Economics and Business, Poznan, Poland
{wlodzimierz.lewoniewski,krzysztof.wecel, witold.abramowicz}@ue.poznan.pl

[2] Aalen University of Applied Science, Business Information Systems, Aalen, Germany
{ralf.haerting,christopher.reichstein}@hs-aalen.de

Abstract. The leading online encyclopedia *Wikipedia* is struggling with inconsistent article quality caused by the collaborative editing model. While one can find many helpful articles with consistent information on Wikipedia, there are also a lot of questionable articles with unclear or unfinished information yet. The quality of each article may vary over time as different users repeatedly re-edit content. One of the most important elements of the Wikipedia articles are references which allow to verify content and to show its source to user. Based on the fact that most of these references are web pages, it is possible to get more information about their quality by using citation analysis tools. For science and practice the empirical proof of the quality of the articles in Wikipedia could have a further signal effect, as the citation of Wikipedia articles, especially in scientific practice, is not yet recognised. This paper presents general results of Wikipedia analysis using metrics from the Toolbox SISTRIX, which is one of the leading providers of indicators for Search Engine Optimization (SEO). In addition to the preliminary analysis of the Wikipedia articles as separate web pages, we extracted data from more than 30 million references in different language versions of Wikipedia and analyzed over 180 thousand most popular hosts. In addition, we compared the same sources from different geographical perspectives using country-specific visibility indices.

Keywords: Data quality · Wikipedia · References · SEO · SISTRIX Sources · Visibility Index · Search engine

1 Introduction

With a range of new technologies like "cloud & mobile computing, social software and big data" and new emerging digital business models, the relevance of services and knowledge databases in the internet like online encyclopedias is increasing rapidly [1–3, 17]. A recent

The original version of this chapter was replaced by an updated version. The correction to this chapter is available at https://doi.org/10.1007/978-3-319-99972-2_49

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 139–152, 2018.
https://doi.org/10.1007/978-3-319-99972-2_11

study investigated [4] that 3.5 billion individuals will have access to internet services and databases in 2017. This number represents 48% of the world population. An ongoing upward trend is expected in the future. Based on this development, the use of Search Engine Optimization (SEO) tools is increasing.

One part of SEO is planning and optimizing of web projects. However, SEO also means monitoring of website performances. In this context, it is possible to distinguish various approaches to control the performance of a web presence [5, 6]. Off-the-page tools that check single performance indicators and especially the visibility rank of hosted websites represent a very effective approach. In this context, there are several enterprises that offer a wide range of specific tools. In digital marketing professional SEO tools are a very widespread application tool segment (e.g. [7, 17]). They are calculating search engine metrics as KPIs (Key Performance Indicators). For our research approach, we used SISTRIX - one of the most popular tool among the European SEO community [17].

One of the good examples of the most visible domains in search engines is Wikipedia. Nowadays, this free encyclopedia is one of the most popular source of information, which is available online. The total number of articles in this collaborative knowledge base is more than 47 million[1]. Wikipedia is also used for enriching other popular knowledge bases (such as DBpedia[2], Wikidata[3] and others). On the other hand, these open data sources are more and more used in various information systems [8, 9].

Due the fact that content of Wikipedia can be freely edited even by anonymous users, quality of this encyclopedia is often subject to criticism[4]. Wikipedia articles often appear among results of search engines (Google, Bing, Yandex and others).[5] There is a temptation to edit the pages to suit specific goals, e.g. spam. This results in vandalism and deteriorating quality. It makes an interesting case for us to study Wikipedia.

Keeping the quality in Wikipedia becomes even more important. Thus, some articles can provide valuable information, especially when considering that there are about 300 language versions in Wikipedia that can be edited at any time and independently. The community of this encyclopedia creates rules that should be followed by authors when writing content. The best articles, which met these criterions, can get special awards for quality. In English Wikipedia such pages are called "Featured Articles" (FA)[6]. However, the number of such articles is very small, e.g. less than 1% in each language version of Wikipedia [10, 11]. Some of the language versions of Wikipedia use more developed quality grading scales whereas articles can receive lower grades, but it requires a user's initiative. As a result, in some language versions of Wikipedia, over 99% of all articles have no grade yet [10, 11]. Additionally, quality of article can change over the time in connection with the continuous edition process of articles and changes of assessment criteria.

1 https://meta.wikimedia.org/wiki/List_of_Wikipedias.

2 http://wiki.dbpedia.org.

3 https://www.wikidata.org.

4 https://en.wikipedia.org/wiki/Criticism_of_Wikipedia.

5 https://www.alexa.com/siteinfo/wikipedia.org.

6 https://en.wikipedia.org/wiki/Wikipedia:Featured_articles.

In addition to the right style of writing, credible and reliable sources of information are one of the most important elements for Wikipedia articles with high quality grades. Users usually have to evaluate the credibility of referenced sources in order to confirm information in an article and assess its quality. Routine work on the evaluation of sources and quality of Wikipedia articles can be supported with automated techniques, which usually take into the account among other only quantity of references without more in-depth analysis. In this study, we propose to use various SEO metrics to analyze the quality of Wikipedia articles and its sources, especially that Wikipedia is becoming more and more quotable for scientific works [12]. Some of these metrics have not been used previously for similar tasks.

We have structured the paper as follows: Sect. 2 introduces the SISTRIX toolbox and the selected indicators. Then we discuss the quality of the Wikipedia articles in Sect. 3, followed by an introduction to the data sets in Sect. 4. Section 5 presents the results of the Visibility Index, while Sect. 6 discusses the results of specific indicators. Section 7 concludes with a summary of the work and an outlook, in particular for further research.

2 Sistrix Toolbox and Selected Indicators

SISTRIX is primarily an analytic tool. Its metrics are very popular in digital marketing and especially in the SEO community [17]. The use of the toolbox provides proposals for further improvements and enables the opportunity to measure and monitor quality aspects of web projects. In general, it is useful to monitor different pages parallel and to analyze the changes of websites over time [13].

Beside common performance indicators for search engine analysis like Backlinks and keyword profile, SISTRIX is also offering more advanced metrics like the Visibility Index and Social Signals related parameters as being explained in the next chapters.

SISTRIX allows to explore the webpage indicators from different levels: domain, host, path, url. Figure 1 shows the example of the webpage from Wikipedia and its different levels.

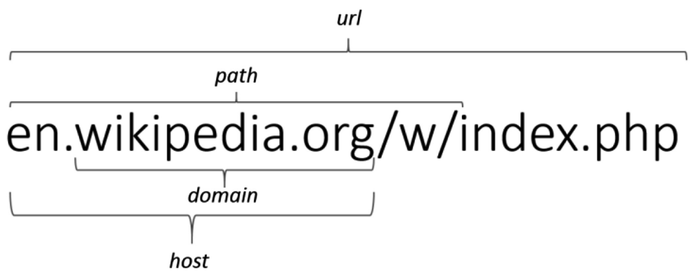


Fig. 1. Domain, host, path and url of the webpage on example of Wikipedia

Visibility Index. The search engine visibility is a well-known and widespread metric within\ SEO tools [14]. Regarding SISTRIX, the key performance indicator (KPI) is calculated based on a dedicated database and shows the visibility of a domain or URL for the search engine result pages (SERP) of Google [7]. The Visibility Index of a domain or URL is usually generated by a keyword pool. There keywords are being ranked and weighed within the Google search results [15]. Every week SISTRIX calculates the index via a keyword pool of one million keywords and keyword combinations. 10% of these keywords are formed by current events. 90% always remain the same. The top 100 positions in Google are analysed and recorded every week. The results are weighted according to the aspects of the position and the expected search volume for each keyword [16]. Similar to Google, SISTRIX keeps the varying algorithm for calculating the Visibility Index secret [17] (Fig. 2).

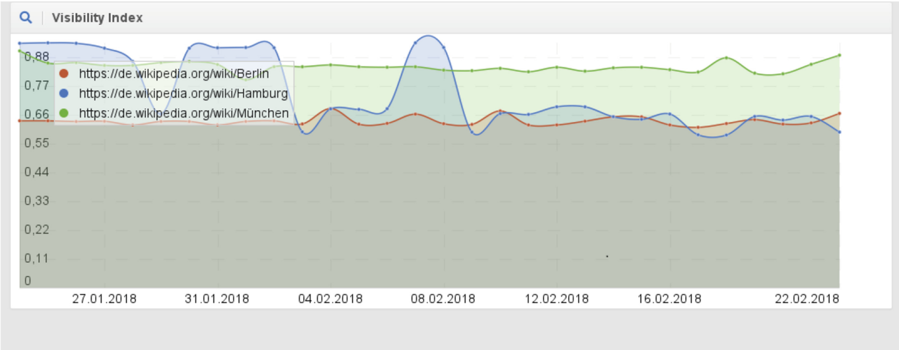


Fig. 2. Comparison of different visibility indices of city articles in Wikipedia [7]

The importance of the Visibility Index for SEO activities receives different assessments [17]. The Visibility Index will not be provided in real-time and cannot provide data about the organic traffic on a website. The second reason is that niche websites with a high number of special keywords will automatically have a lower Visibility Index. Referring to [17], it can be assumed that there is a correlation between the Visibility Index and the quality of a website because the behavior of the Google algorithm is a relevant factor of the SISTRIX Visibility Index. The Google search engine ranks websites with high-quality content and relevance with higher priority. Therefore, quality can have an effect on the Visibility Index. It remains the problem that websites focusing on niche content and keywords are sometimes not listed in the database of SEO-Tools, even if they are of good quality [18].

Number and Profile of Keywords (SEO). The Number of Keywords shows how many different keywords the domain (host or URL) has achieved a ranking in the Top-100 SERPs. For that indicator SISTRIX is using the abbreviation “SEO” [7]. An extended consideration of the distribution of keywords leads to a Keyword Profile. The Keyword Profile is part of the SEO Overview section in SISTRIX Toolbox. It enables a deeper understanding of the keyword structure on a quantitative basis. In SEO, keywords are one important factor in terms of on-page optimization. Especially the position of the keyword (e.g. in H1 tag), contributes much to the Google ranking. This indicates that Google classifies pages dedicated to a specific topic or keyword as more relevant. Under that aspect the position in the Google results, which is measured by the SISTRIX keyword profile, could indicate the relevance of a page. In that context, it can be assumed that the results of SISTRIX Keyword-Profiles also show a clear gradation regarding the quality of Wikipedia articles.

Universal Search. Websites that are listed in the first ranks of the SERP have a higher chance of generating web traffic [17]. Current research shows in particular the relevance of optimization based on Universal Search. Google has been gathering much more information than plain text content from websites [19]. Universal Search goes a step further and, in addition to plain text oriented websites, currently includes in the ranking formats like news, business databases, maps, videos or scanned books from Google Books. Depending on the format type, the Universal Search hits are integrated into the familiar layout of the SERPs. The click-through-rate in the SERPs has shifted because Universal Search results are often preferred even if they are on page two or three of the SERPs. In Sistrix the Universal Search metric reflects the integration of these formats on the 1st page of the SERPs. Derived from the indicator "keyword profile" we can finally state an impact on the quality of Wikipedia articles.

Backlinks. A Backlink is defined as an external link that refers back to the domain under consideration [20]. The number of Backlinks counts the number of in-links from a certain website that appear within the World Wide Web [29]. Before search engines like Google and Yahoo became of such a huge significance, Backlinks have been one of the main means for web navigation. With the upcoming of search engines, the quantity of Backlinks became a major factor to analyze the popularity and importance of a website [21]. Today Backlinks are still important for SEO but they are focusing more on the quality of a link and not on the quantity anymore [18]. It is possible that high-quality Backlinks influence the quality of Wikipedia articles, because on the one hand, Google (as the most important search engine in the world) highly weights Backlinks and on the other hand, links to trustworthy domains are usually a good indicator of quality.

Social Signals (Facebook, Twitter, Google+). Since inception of Facebook or other social networks, Social Signals also became an important information for SEO. They provide information about social interactivity, social behavior and social relations [18]. Social Signals can be comments, likes or shares on Facebook, for instance. In SISTRIX, the Social Signal indicator measures the type and quantity of Social Signals available for URLs of a domain. SISTRIX covers the five large social networks: Facebook, Twitter, Google+, LinkedIn and Pinterest. The correlation between Social Signal and the ranking position of a URL is extremely high. This is valid for all social networks covered by SISTRIX and leads to the assumption that the results of Social Signals also correlate with the quality of Wikipedia articles.

To investigate the quality of Wikipedia articles, we used aforementioned indicators of the SISTRIX Toolbox: Visibility Index, Number of Keywords (SEO), Universal

Search, Backlinks and Social Signals. Via the API of SISTRIX, we extracted data from 189.800 hosts up to 25 countries. Based on the queries we then compared results of different countries. In top 8 countries (United States, Germany, France, Italy, Poland, Great Britain, Spain and Austria), the language versions of reference hosts have been most frequently queried. We used for the investigation 7 country-specific versions of the SISTRIX Toolbox related to Germany (DE), Spain (ES), France (FR), Italy (IT), Poland (PL), United Kingdom (UK) and United States (US). Based on the language mix of SISTRIX versions, we generated a multidimensional perspective. Several indices are available for a single reference host.

3 Quality in Wikipedia

Wikipedia has a lot of articles without quality grade. Automatic quality assessment of Wikipedia content is sufficiently developed topic. Generally, scientific works in this direction are concentrated on the analysis of various content metrics and articles history analysis.

Many studies showed so far that generally best articles in Wikipedia were usually characterized by a longer text, bigger number of references, sections, images and other content metrics [10, 11, 22, 23]. Online service *WikiRank*[7] used some of the metrics to compare the quality of Wikipedia articles between various language versions. Second important aspect was popularity of articles, which could also be correlated with the number of edits [24]. One of the studies showed that popular articles with bigger number of edits are located closer to the top of the knowledge classification hierarchy [25]. Authors of other studies concluded that popular topics in Wikipedia had a larger number of articles with high quality [26]. In this study, we decided to analyze the suitability of SEO-related metrics for determination of quality and popularity of Wikipedia articles.

One of the most important content metrics in quality assessing of Wikipedia articles is the number of references [10, 22]. The more challenging task is to analyze not only quantity, but also quality of references. However, researches related to the analysis of references are not developed enough. One of the studies showed journal's impact factor predicts its appearance as a source in Wikipedia articles [12]. That work also showed, that accessibility of these journals could increase probability of referencing on Wikipedia as well. Other researches concentrated on Wikipedia articles related history [27]. We found that often sources of such articles were mass media, US government websites and also some references to academic journals. However, those and other studies took only English into account - as one of the most developed language version of Wikipedia. Other work concentrated on similarity analysis of the references [28], which could be useful for comparing different language versions of articles.

Finally, we did not find the scientific works that assessed quality of Wikipedia content based on qualitative analysis of the references. Therefore, we decided to use

7 http://wikirank.net.

some SEO metrics to analyze the sources of the articles in Wikipedia in order to get preliminary results for assessing future work potential.

4 Datasets

For our study we chose six Wikipedia language editions: English with over 5 million articles, German with over 2 million articles, and French, Spanish, Italian, Polish with over million articles each.

In order to analyze how metrics can affect the quality of Wikipedia, we divided the research into two parts focusing on different aspects:

1. References with URLs contained in Wikipedia articles.
2. Wikipedia articles as separate web pages.

4.1 Referenced Hosts

Using own improved algorithms (parsers) from previous study [28], we extracted data from references of over 10 million articles of the considered language versions of Wikipedia based on dumps from November 2017. The methods consisted in searching references with URL addresses in "ref" tag in source code of the articles. Additional information was extracted from special citation templates[8], which can have different names and parameters in each language version. Then, we extracted unique hosts from all URL addresses of these references. Table 1 shows the numbers of extracted URL addresses of references and number of unique hosts.

Table 1. Number of references with URLs and number of hosts

Language edition	Number of references with URL	Number of hosts
English Wikipedia	26 200 779	1 916 682
French Wikipedia	4 275 477	460 402
German Wikipedia	4 139 912	608 151
Spanish Wikipedia	3 314 033	380 697
Italian Wikipedia	2 652 314	288 428
Polish Wikipedia	2 270 715	222 434

We chose 50,000 of the most popular hosts as used in references in each considered language versions of Wikipedia. As it can be expected, there are hosts, which are common for some languages. Figure 3 shows overlaps between obtained lists of the top hosts. Before analyzing theses hosts in SISTRIX, we combined these lists and removed duplicates, receiving finally 189800 hosts.

[8] https://en.wikipedia.org/wiki/Help:Citation_Style_1.

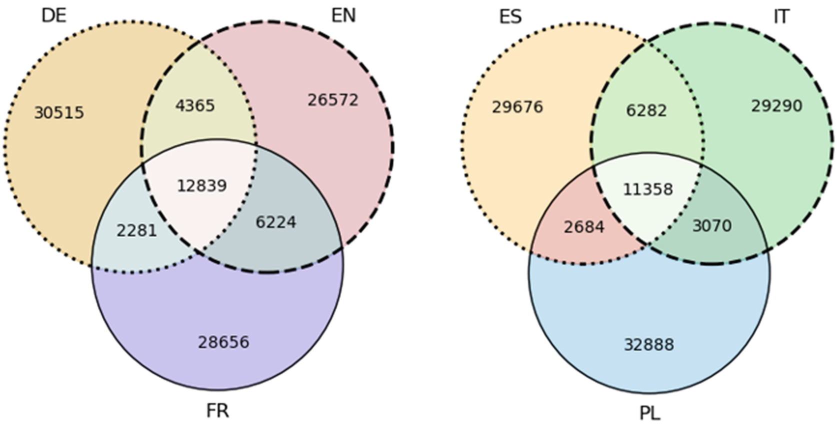


Fig. 3. Overlaps in lists of the most popular hosts (top 50000)

4.2 Wikipedia Articles

The measurement for different indicators of Wikipedia articles was conducted in February 2018 using Sistrix Toolbox. As mentioned before, Wikipedia articles can receive various quality grades from users. However, each language edition of Wikipedia can has own systems of grades, which can be more or less developed. That's why we chose the English Wikipedia, which is the largest language edition of Wikipedia and has also a well-developed assessment system. The English Wikipedia has the biggest number of articles with defined quality grades. Therefore, we decided to take the best articles (FA) and the articles with the lowest quality grade (stub) from this language version. Additionally, we divided these articles into two groups: the most popular and the less popular articles. Thus, the four groups of Wikipedia articles are distinguished:

- PFA – Popular Featured Articles,
- UFA – Unpopular Featured Articles,
- PST – Popular Stub Articles,
- UST – Unpopular Stub Articles.

For each group, we selected 100 articles and extracted them from Wikipedia dumps[9] from February 2018, and page visits from February 2017 till January 2018 (for selecting popular articles). From about 5.2 thousand of FA articles, we chose 100 most popular (PFA) and 100 least popular ones (UFA). We repeated the procedure for about 2.3 million of Stub articles (PST and UST). It has to be noticed that higher quality did not result in a higher popularity. In this context, the most popular Stub article might have a slightly larger popularity than the most popular FA article, for instance.

[9] https://dumps.wikimedia.org.

5 Visibility Index Results

5.1 References Analysis

In order to analyze Visibility Index of the references in each language version of Wikipedia, we selected only hosts that had defined metrics for one of the 7 countries: Germany (DE), Spain (ES), France (FR), Italy (IT), Poland (PL), United Kingdom (UK), United States (US). This choice was made in order to analyze how depending on the language version of Wikipedia, the regional (local) values of Visibility Index differ. Table 2 shows that references from local hosts (same language) more often have nonzero Visibility Index.

Table 2. Shares of hosts from top 50000 with nonzero Visibility Indexes in each Wikipedia language editions from different countries perspectives

Wikipedia edition	Countries						
	DE	ES	FR	IT	PL	UK	US
German	.524	.191	.226	.172	.147	.314	.303
English	.342	.343	.354	.278	.232	.615	.603
French	.210	.207	.541	.177	.136	.336	.320
Italian	.214	.213	.224	.467	.140	.348	.334
Spanish	.214	.492	.241	.198	.147	.368	.376
Polish	.205	.180	.187	.158	.429	.305	.298

Table 3 shows that values of Visibility Index of regional hosts are more stable in own local languages (lower standard deviation).

Table 3. Standard deviation of hosts from top 50000 with nonzero Visibility Indexes in each Wikipedia language editions from different countries perspectives

Wikipedia edition	Countries						
	DE	ES	FR	IT	PL	UK	US
German	52.90	91.51	67.23	87.21	68.71	89.39	81.79
English	77.45	102.93	63.99	82.75	48.32	77.31	69.79
French	78.44	105.29	44.49	82.50	49.86	83.34	76.39
Italian	78.28	105.33	65.85	53.40	49.56	82.81	75.58
Spanish	80.79	71.73	66.14	81.23	68.38	82.66	73.16
Polish	80.99	114.33	70.92	88.89	46.79	88.94	80.41

Median values of the Visibility Index for each sets of hosts also have bigger values in local languages. Table 4 presents the results of such dependency.

Table 4. Median of hosts from top 50000 with nonzero Visibility Indexes in each Wikipedia language editions from different countries perspectives

Wikipedia edition	Countries						
	DE	ES	FR	IT	PL	UK	US
German	.09	.05	.05	.04	.04	.08	.08
English	.04	.05	.05	.04	.03	.08	.09
French	.04	.05	.07	.04	.03	.07	.07
Italian	.04	.05	.05	.09	.03	.08	.08
Spanish	.04	.08	.05	.04	.03	.08	.08
Polish	.05	.06	.06	.05	.09	.08	.08

5.2 Wikipedia Articles Analysis

As with the references, we extracted the Visibility Index for each Wikipedia article from different countries perspectives. Table 5 presents mean values of this metric for each articles group and country.

Table 5. Mean of Visibility Index each articles category from different countries perspectives

Wikipedia edition	Countries						
	DE	ES	FR	IT	PL	UK	US
PFA	.059	.037	.041	.015	.026	.255	.234
UFA	.000	.000	.000	.000	.000	.000	.000
PST	.002	.000	.003	.001	.001	.020	.020
UST	.000	.000	.000	.000	.000	.000	.000

The results show that articles with high visibility indices are relatively more popular. This can be explained by the fact that higher visibility in search engines mean additional visitors. A interesting fact is that for high quality FA articles this metric has much larger values than Stub. This is despite the fact that popularity of articles in PFA group do not differ much from PST.

6 Other Indicators Results

6.1 SEO, Universal-Search, Backlinks of the Hosts

For references, we additionally extracted three indicators: SEO, Universal Search and Backlinks (or Links). Figure 4 shows that for German Wikipedia references hosts have relatively higher values for SEO and Universal Search, while English has the highest values of Backlinks metrics.

Links can be further investigated by country. In our hosts dataset we get links from over 200 different countries. Next figure presents distribution of this indicator among

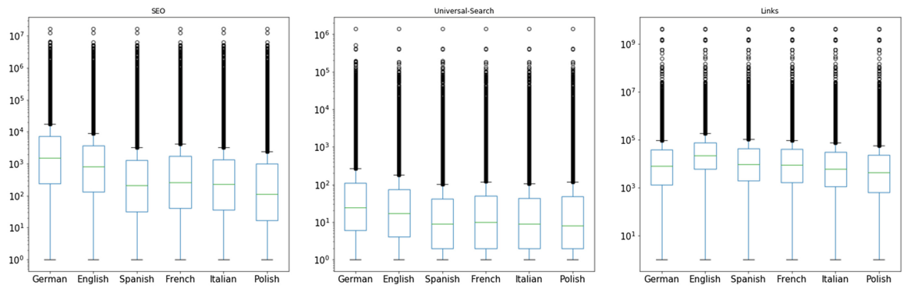


Fig. 4. Distribution of indicators values of references hosts in selected language version of Wikipedia. Source: own calculations in November 2017

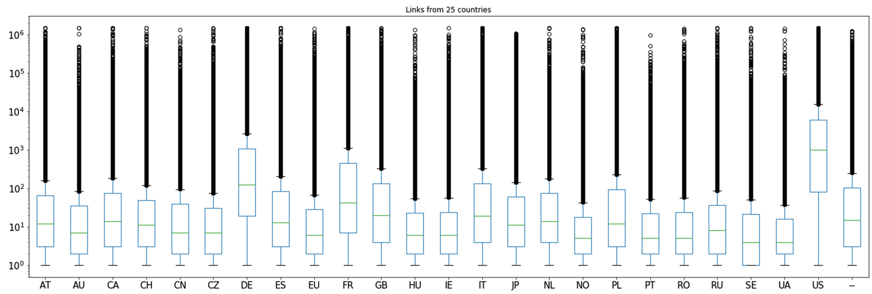


Fig. 5. Distribution of Backlinks indicator values of references hosts from different countries

all considered popular references hosts from 25 countries[10] ("--" - means that the country is not defined), which have the largest number of the links (Fig. 5).

In the top 8 of more popular countries there are: United States (US), Germany (DE), France (FR), Italy (IT), Poland (PL), Great Britain (GB), Spain (ES), Austria (AT). In these countries, the most frequently used are the language versions that we are exploring.

6.2 Social Signals of Wikipedia Articles

Using Sistrix Tool we obtained following socials indicators for Wikipedia articles dataset: FB - all signals from Facebook, FBl - Facebook likes, FBs - Facebook shares, FBc - Facebook comments, TW - signals from Twitter, LI - signals from LinkedIn, GP - signals from Google Plus, PT - signals from Pinterest. Table 6 presents mean values of each considered social indicator in each articles group. As in the case with the Visibility Index, the popular articles has bigger values of social metrics, but the value are much higher in articles of better quality.

[10] Extended results can be found under http://data.lewoniewski.info/bis2018seo/.

Table 6. Mean of each social indicators for each Wikipedia articles group. Source: own calculation in February, 2018.

Articles	Social signals							
	FB	FBl	FBs	FBc	TW	LI	GP	PT
PFA	2101.7	1138.0	571.9	391.8	28.1	16.9	286.4	174.5
UFA	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
PST	5.6	2.4	1.9	1.2	0.4	0.2	0.6	0.0
UST	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

7 Conclusion and Future Work

In this study, we showed results of analysis of different SEO indicators related to Multilingual Wikipedia and its references: Visibility Index, SEO, Universal-Search, Backlinks and Social Signals.

The results showed that popular articles in Wikipedia have higher visibility indices and social indicators. Values of such metrics are significantly higher if articles have FA grade. By analyzing references, we discovered that often SEO indicators for respective pages are higher in countries that are related to their Wikipedia language version. We also studied the most popular countries that are often linked to Wikipedia references hosts. We have found that in the top 8 countries, all of them were strongly related to considered language versions.

For science and practice alike, it can be concluded that the quality of the articles increases due to the large number of references given in the examined Wikipedia articles. Even if the quality was only measured quantitatively, it can still be said that a large number of Wikipedia articles can be regarded as citable from the authors' point of view. A qualitative approach could certainly be sought for further research.

In future, we plan to include additional indicators, expand number of considered articles and language versions. Our goal is to discover how these indicators are connected with various quality metrics of articles. Some of the indicators can be also used in detection of language sensitive topics in Wikipedia. Considered metrics can help to analyze quality of particular parts of articles - infoboxes. There is a project Infoboxes[11], which compares quality of these parts between various language versions of Wikipedia. In this context, a comparison with other SEO tools, especially for corporate practice, would also be a great advantage. The associated work could be used to confirm the selection of different metrics.

Future work may also encompass analysis of metrics on various granularity: host-level, domain, path and url. Some of the SEO metrics can affect the popularity of websites that serve as a source in Wikipedia articles. If a domain has better SEO metrics, Wikipedia users can use this website more often for references. SEO metrics (especially Visibility Index) change over the time. In comparable periods, the relevance of a source as a reference in Wikipedia articles may also vary.

[11] http://infoboxes.net.

Existing and new metrics can help in future for more complete analysis of the quality of content in various language versions of Wikipedia and can allow to choose the best ones. Less developed language editions of Wikipedia can be enriched automatically. Metrics can also help to improve quality in DBpedia, Wikidata and other open knowledge bases [11].

References

1. Gantz, J., Reinsel, D.: The digital universe in 2020: big data, bigger digital shadows, and biggest growth in the far east (2012). http://www.emc.com/collateral/analyst-reports/idc-the-digital-universe-in-2020.pdf
2. Bughin, J., Chui, M., Manyika, J.: Clouds, big data, and smart assets: ten tech-enabled business trends to watch. McKinsey Q. **56**(1), 75–86 (2010)
3. Schmidt, R., Möhring, M., Härting, R.-C., Reichstein, C., Neumaier, P., Jozinović, P.: Industry 4.0 - potentials for creating smart products: empirical research results. In: Abramowicz, W. (ed.) BIS 2015. LNBIP, vol. 208, pp. 16–27. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19027-3_2
4. International Telecommunication Union: Measuring the Information Society Report 2017, vol. 1 (2017). https://www.itu.int/en/ITU-D/Statistics/Documents/publications/misr2017/MISR2017_Volume1.pdf
5. Kumar, L., Kumar, N.: SEO technique for a website and its effectiveness in context of Google Search Engine. Int. J. Comput. Sci. Eng. (IJCSE) **2**, 113–118 (2014)
6. Schroeder, B.: Publicizing your program: website evaluation, design, and marketing strategies. AACE J. **15**(4), 437–471 (2007)
7. SISTRIX GmbH: The secret of successful Websites. http://www.sistrix.com
8. Stróżyna, M., Eiden, G., Abramowicz, W., et al.: A framework for the quality-based selection and retrieval of open data - a use case from the maritime domain. Electron Mark. (2017). https://doi.org/10.1007/s12525-017-0277-y
9. Filipiak, D., Filipowska, A.: Improving the quality of art market data using linked open data and machine learning. In: Abramowicz, W., Alt, R., Franczyk, B. (eds.) BIS 2016. LNBIP, vol. 263, pp. 418–428. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-52464-1_39
10. Lewoniewski, W., Wecel, K., Abramowicz, W.: Relative quality and popularity evaluation of multilingual Wikipedia articles. Informatics **4**, 43 (2017)
11. Lewoniewski, W.: Enrichment of information in multilingual Wikipedia based on quality analysis. In: Abramowicz, W. (ed.) BIS 2017. LNBIP, vol. 303, pp. 216–227. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-69023-0_19
12. Teplitskiy, M., Lu, G., Duede, E.: Amplifying the impact of open access: Wikipedia and the diffusion of science. J. Assoc. Inf. Sci. Technol. **68**(9), 2116–2127 (2017)
13. Drèze, X., Zufryden, F.: Measurement of online visibility and its impact on Internet traffic. J. Interact. Mark. **18**(1), 20–37 (2004)
14. Goodman, A.: Winning Results with Google AdWords, 2nd edn. McGraw-Hill, New York City (2009)
15. Maynes, R., Everdell, I.: The Evolution of Google Search Results Pages & Their Effects on User Behaviour (2014). http://www.mediative.com/whitepaper-the-evolution-of-googles-search-results-pages-effects-on-user-behaviour/
16. Kronenberg, H.: Wie wird der Sichtbarkeitsindex berechnet? (2013). http://www.sistrix.de/frag-sistrix/was-ist-der-sistrix-sichtbarkeitsindex/

17. Härting, R.-C., Mohl, M., Steinhauser, P., Möhring, M.: Search engine visibility indices versus visitor traffic on websites. In: Abramowicz, W., Alt, R., Franczyk, B. (eds.) BIS 2016. LNBIP, vol. 255, pp. 91–101. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39426-8_8
18. RYTE GmbH: Search Engine Optimization (2018) https://en.ryte.com/wiki/Category:Search_Engine_Optimization
19. Berman, R., Katona, Z.: The role of search engine optimization in search marketing. Mark. Sci. **32**(4), 644–651 (2011)
20. Searchmetrics: Backlinks Definition - SEO Glossary. https://www.searchmetrics.com/glossary/Backlinks/
21. Killoran, J.B.: How to use search engine optimization techniques to increase website visibility. IEEE Trans. Prof. Commun. **56**(1), 50–66 (2013)
22. Warncke-Wang, M., Cosley, D., Riedl, J.: Tell me more: an actionable quality model for Wikipedia. In: Proceedings of the 9th International Symposium on Open Collaboration, p. 8. ACM (2013)
23. Wecel, K., Lewoniewski, W.: Modelling the quality of attributes in Wikipedia infoboxes. In: Abramowicz, W. (ed.) BIS 2015. LNBIP, vol. 228, pp. 308–320. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-26762-3_27
24. Peinado, A.J.R., Barahona, J.M.G.: Temporal and behavioral patterns in the use of Wikipedia. Doctoral dissertation, Ph.D. thesis, Universidad Rey Juan Carlos, pp. 128, 139 (2011)
25. Lerner, J., Lomi, A.: Knowledge categorization affects popularity and quality of Wikipedia articles. PLoS ONE **13**(1), e0190674 (2018)
26. Lehmann, J., Müller-Birn, C., Laniado, D., Lalmas, M., Kaltenbrunner, A.: Reader preferences and behavior on Wikipedia. In: Proceedings of the 25th ACM Conference on Hypertext and Social Media, pp. 88–97. ACM (2014)
27. Luyt, B., Tan, D.: Improving Wikipedia's credibility: references and citations in a sample of history articles. J. Assoc. Inf. Sci. Technol. **61**(4), 715–722 (2010)
28. Lewoniewski, W., Wecel, K., Abramowicz, W.: Analysis of references across Wikipedia languages. In: Damaševičius, R., Mikašytė, V. (eds.) ICIST 2017. CCIS, vol. 756, pp. 561–573. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67642-5_47
29. Klusch, M.: Information agent technology for the internet: a survey. Data Knowl. Eng. **36**(3), 337–372 (2001)

How to Describe Basic Urban Pattern in Geographic Information Systems

Indraja E. Germanaitė[1(✉)], Rimantas Butleris[2], and Kęstutis Zaleckis[3]

[1] Department of Information Systems, Kaunas University of Technology, Kaunas, Lithuania
indraja.germanaite@ktu.lt
[2] Centre of Information Systems Design Technology, Kaunas University of Technology, Kaunas, Lithuania
[3] Faculty of Civil Engineering and Architecture, Kaunas University of Technology, Kaunas, Lithuania

Abstract. Spatial patterns play an important role in the Spatial Data Analysis performed by Geographic Information Systems. This paper presents the analysis of the urban pattern description in the form of UML class diagrams covering the aspects of the hierarchy and generalization of patterns and metapatterns. In addition, the data model for keeping the 3D geometric and topographic data of the urban pattern is reviewed. Subsequently, the article presents a survey of the methods and solutions of spatial analysis, concentrating on the methods based on space syntax, which could be used in further research and computerization of the methodology of urban patterns for Geographic Information Systems.

Keywords: Pattern · Space syntax · Geographic Information Systems

1 Introduction

Christopher Alexander [1] presented that purpose of the pattern analysis is to detect invariant things in the infinite set of all possible variations of the real life urban architecture [1]. He also created the pattern language and the elements of this language, entities called patterns [2]. The pattern describes a problem which occurs over again in the environment and the core of the solution to that problem [2].

During the last decade, much effort has been put into developing theories and methods of Spatial Data Analysis (SDA) to improve understanding and modeling of real-world phenomena [3]. Recent research and applications in SDA have developed many sophisticated analytical techniques for studying spatial patterns and processes [3]. The discovery and the recognition of spatial and urban patterns are among the most popular tasks for most GIS applications and plug-ins.

Though each Alexander's pattern has the same format: an example of the pattern (a picture), the context of the pattern (text paragraph), the problem and the solution (a description, an instruction and a diagram), and the connection to all other smaller patterns in the language (a list) [2], but all patterns are very different in complexity, hierarchical level, attributes and spatial features. There is a need to develop a methodology of describing and applying various spatial patterns that could be used

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 153–163, 2018.
https://doi.org/10.1007/978-3-319-99972-2_12

with SDA methods in GIS. For example, spatial patterns could be used to detect how urban pattern correlates and affects urban crime [4] or to examine morphological changes and the transformations of the urban settlement [5].

An urban pattern may be regarded as a complex physical entity (such as a road) or as any kind of structure, spatial distribution, or a recurring feature (such as a street pattern) that is represented as lines, areas, or bodies in a 2D or 3D map [6].

There are quite many ways for describing or measuring patterns, such as Qualitative methods, which include morphological and morphographic descriptions, and Quantitative methods, which include network component analysis, graph theory (e.g. space syntax), and the fractal dimension [6]. The identification of urban pattern types, properties, and indicators could be performed by using urban geometry – the geometric interpretation of urban geography - and urban topology – the mathematical study of configurations and relations between entities independently of their absolute (metric) dimensions [6].

Even though urban patterns can be regarded as abstract types that allow generalization, there is no single correct or definitive way of classifying patterns or identifying pattern types, and a diversity of overlapping types and themes are both appropriate and inevitable [6].

There are two aims of this paper: (1) to describe a very basic urban pattern and data model that could be later improved and used in spatial analysis of vector maps and (2) to overview some of the methods and solutions of spatial analysis, concentrating on the methods based on space syntax, and to extract basic components of those methods that could be used in further research and computerization of the urban patterns methodology.

2 How to Describe an Urban Pattern

Spatial objects are usually easily distinguished and named discrete and bounded entities [7]. The space between spatial objects is potentially "empty" or undetermined, and the position of the objects in space is determined by a combination of one or more of the following parameters: location, shape, size, and orientation [7]. As shown in the classification of spatial objects [8], a spatial object is composed by at least one dimensional element. For a simple and composed spatial object, each dimensional element is joined to all the other dimensional elements; however, there is also a complex spatial object (as shown in Fig. 1) [8], which can contain disconnected dimensional elements.

Fig. 1. Simple, composed and complex spatial objects [8]

The urban pattern could be constructed using small indivisible spatial or logical elements (metapatterns) that later could be used to identify patterns of different scale and complexity. The term metapattern means a pattern of patterns [9]. It is possible to choose any pattern classification system that serves for the application purposes as long as it uses non-complex elements that have some topological expression. We called this topological expression *Shape* in the class diagram of the pattern structure shown in Fig. 2.

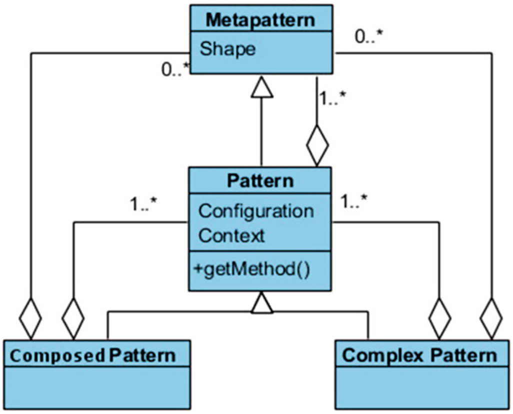


Fig. 2. Pattern class diagram

A pattern is constructed of at least one metapattern, and this construction is called *Configuration*. Composed and Complex patterns both can be assembled from at least one pattern and zero or more metapatterns. This was done with the intention of distinguishing (a) indivisible spatial elements (called metapatterns) and (b) quite simple yet composite patterns that are constructed of several metapatterns (called patterns). Both Composite and Complex patterns can use more than one pattern and zero or more metapatterns. These two differ only by how the elements (metapatterns and patterns) are connected: for a Composed pattern, all elements must be joined, but for a Complex pattern, this is not mandatory. A class *Pattern* also has an operation *getMethod()*, which indicates that a *Pattern* should have a method for detecting physical patterns in spatial vector-based data, but the detailed description of this method is outside the scope of this paper. For now, such spatial object characteristics like location, size, and orientation, which are absolute (metric), will not be used either.

For the illustration of the Pattern class diagram shown in Fig. 2, we will add some urban data and take two metapatterns – Center [9] and Border [9] to construct a Core–periphery pattern [10]. Then, using these three elements, we can make three Composed patterns – a Concentric zone [11], a Sector [11], and a Linear [11], and one Complex pattern – a Multi-nuclei [11] by mixing the already defined Core–periphery pattern with some metapatterns – Tube [9] and Sheet [9]. The illustration of the city patterns is presented in Fig. 3, and the Urban pattern class diagram with the generalization of the urban patterns is presented in Fig. 4.

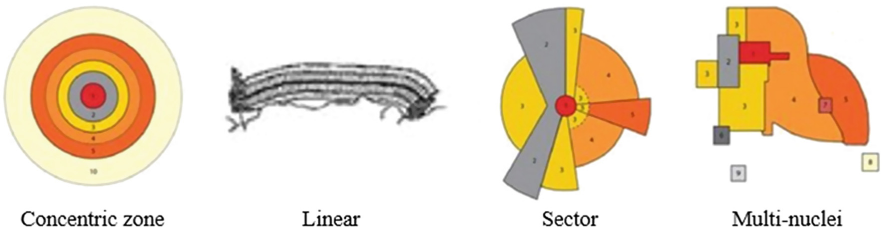


Fig. 3. City patterns [11]

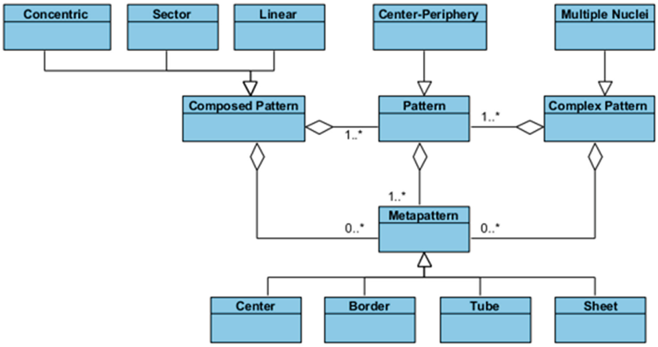


Fig. 4. Urban pattern class diagram with generalizations

It is important to notice that different urban patterns exist on different hierarchical levels of the city structure. At the same time, the same urban pattern can also be found on different hierarchical levels of the city structure, e.g. in the building, in the group of buildings, in the suburb area, in the whole city, or even in the whole region. Thus, the *Pattern* class also has the *Context* attribute. Other attributes that would facilitate the identification and would help solving the issues with the transformation and application of the urban pattern still need to be defined.

3 How to Describe Urban Pattern Data Model

There is a growing interest in 3D software programs that include changes from computerized 2D graphic design software to 3D sketching software as well as in programs for the planning of spaces and buildings in virtual spaces, analytical maps (Quantum GIS), and Google Earth [12]. The research of 3D city models (3DCM) is a multidisciplinary study field of urban GIS, Computer-Aided Design (CAD), and presentation and spatial modeling, and has been considered as one of major directions of spatial information research [13]. In GIS applications for Transportation (known as GIS-T) or Utility management, a network is typically described using a node-arc model based on a planar graph [13]. However, the node-arc model cannot represent some real features or analyze topological relationships of networks with other urban objects in 3D [13].

GIS researchers have extended 3DCM 3D data models to represent geo-referenced objects and are interested in algorithms and structures for processing large 3D data in real time as well as in photo-realistic visualization and extended VR tools, but the corresponding work in the GIS community is directed to more fundamental levels, i.e. the development of models for maintaining 3D topology as the basis of 3D GIS [13].

A spatial object in general consists of the aspects theme, geometry, topology, and time [14]. The information on the modeled domain is usually separated into models of geometric space (2D/3D) and thematic aspects [8].

Thus, the description of an urban pattern should also be suitable for at least 3D space, and time dimension could also be discussed - for example, when there is a need to analyze historical data and gather knowledge on how the object changed over time. In addition, an urban pattern should support thematic, geometric, and topological data.

The structures describing the thematic aspects of the spatial objects could be realized using an object-oriented model [14]. Traditional CAAD programs internally represent data using geometric entities such as points, lines, rectangles, and planes. Thus, they cannot capture domain-specific information about entities. To overcome the limitations of general-purpose geometric representations, researchers have been developing and using object-based data models that are specific to their domain [15].

The development of the data model for 3D geometry is largely influenced by the model of Molennar (shown in Fig. 5 [16]), which combines the geometry and topology of 3D spatial data and allows for the retrieval of multiple topological properties directly from the model [14]. The basic concepts include the primitive node (point), arc (line), and face (area), and extend the earlier models by the new primitives - edge and body for a 3D model [14]. The topology of the 3D primitives has been modeled through several 1:n relationships between the five primitives [14]. Thematic attribute data are attached using feature identifiers [14].

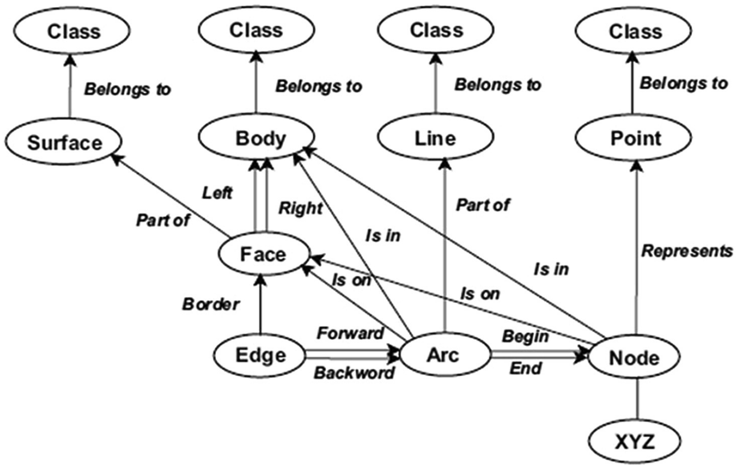


Fig. 5. 3D Formal Data Structure (3DFDS) (Molennar) [16]

There are multiple 2D, 3D, and 4D data models for keeping and maintaining spatial data, which extended the 3DFDS model in various ways, but for the initial purpose of describing an urban pattern, the plain 3DFDS model was chosen for presenting the main principles. It is worth researching the advantages of other models that could be used for a specific project or thematic tasks and should be explored in the future.

Based on topological relationships between 3D primitives [14], a simple 3D topological information class diagram for the use of the urban pattern is shown in Fig. 6.

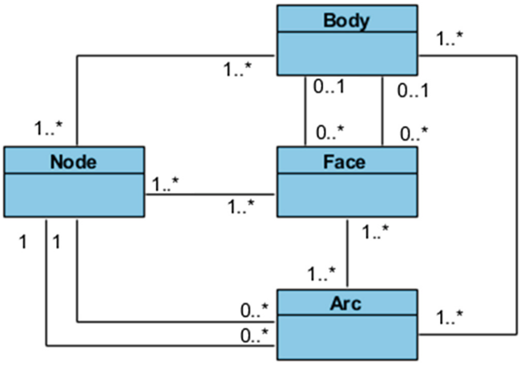


Fig. 6. 3D urban pattern topological information class diagram (using topological relationships discussed in [14])

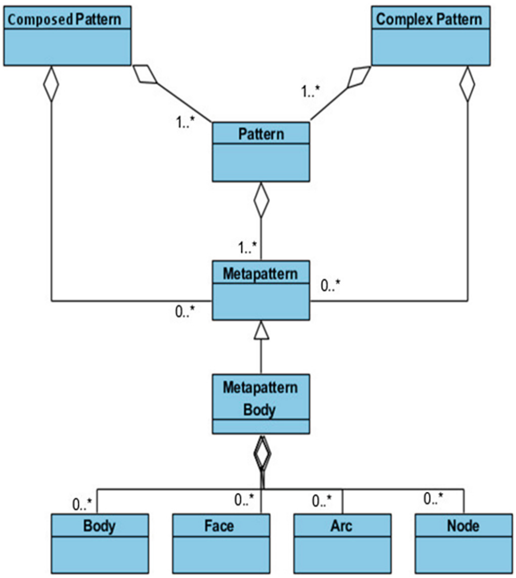


Fig. 7. 3D urban pattern class diagram

A completed 3D urban pattern class diagram is shown in Fig. 7. It combines the pattern (shown in Fig. 2) and 3D topological (shown in Fig. 6) information.

4 How to Calculate Urban Pattern

With the basic description of the urban pattern and with the basic geometry and topology notion, the methods how to detect and calculate the real urban pattern can be chosen. Urban pattern calculation method, called *getMethod()*, belongs to *Pattern* class as presented in Fig. 2. In this part the short overview of some existing spatial analysis methods and solutions, concentrating on the methods based on space syntax, is done.

Spatial analysis is the generic term for all manipulations of spatial data carried out to improve understanding of the geographic phenomena that the data represents through discovering patterns that were previously unknown, or to build arguments on which to base important decisions [7]. A large number of methods for the analysis of the spatial structure of natural phenomena have been developed in a wide range of scientific fields [19]. Main groups of analytical GIS capabilities are: (1) Classification, retrieval and measurement functions, that are performed on a single (vector or raster) data layer; (2) Overlay functions, that allow the combination of two (or more) spatial data layers and treating areas of overlap and of non-overlap in distinct ways; (3) Neighborhood functions, that evaluate the characteristics of an area surrounding a given feature's location; (4) Connectivity functions, that work on basis of networks, that represent spatial linkages between features [7].

4.1 Computational Space Syntax Model and Algorithms

Space syntax as a method could be used to detect at least some of the urban patterns in spatial data. Space syntax is based on topology and graph theory as well as on quantitative analysis and geospatial computer technology and provides a set of theories and methods for the analysis of spatial configurations of all kinds and at all scales [11, 14]. The computational space syntax model is based on a graph-oriented representation of a geographical space and is based on a two-step approach: (1) to represent the large-scale space as a finite number of small-scale spaces; (2) to link these individual small-scale spaces to form a connectivity graph [7].

Several authors have developed different computer programs for the construction of axial and segment maps: some programs have been implemented in GIS, such as Axman (a Mac-based application) and Axwoman (a Windows-based application), both used to draw axial lines and to analyze axial maps of urban and interior spaces [17]. Other software programs have been designed to generate axial and segment maps automatically - for, instance, a universal-platform software called Depthmap used to perform a set of spatial network analyses designed to explain social processes within a built environment, or AxialGen - a research prototype for automatically generating axial maps to demonstrate that axial lines constitute a true skeleton of an urban space [17]. The most recent version of Depthmap now supplies a range of configurational analyses that include the original visibility analysis, the generation and analysis of axial maps, and segment analysis [18].

The Space Syntax Toolkit (SST) is a Python plug-in for QGIS that integrates depthmapX[net] and is used for spatial network and statistical analysis. The SST implements the basic workflow of axial and segment map analysis from map production to the publication of the results, emulating in part the exploratory spatial analysis features of depthmapX. It is primarily aimed at supporting the space syntax methodology and enhancing it with GIS data, analysis, and visualization [19].

Urban Network Analysis (UNA) toolbox is an open-source toolbox for spatial network analysis in the ArcGIS 10 software platform. It introduced two important modifications to the network representation of the built environment. First, it added buildings (or other location instances) to the representation, adopting a tripartite representation that consists of three basic elements: edges, nodes, and buildings [20]. Second, the UNA toolbox introduced a weighted representation of spatial network elements: each building obtains a set of attributes that connect the building in the graph with the true characteristics of the corresponding structure in the city [20].

Computational algorithms to evaluate design solutions using space syntax during the process of computer-aided architectural designing were also developed [15]. These algorithms extract topological information from design solutions and recognize building information produced in the form of Industry Foundation Classes (IFC), to deduce the necessary topological information and to store the information in the form of matrices (two-dimensional arrays) [15].

4.2 Other Methods of the Spatial Analysis

Space syntax methods and metrics are not the only ones to use in spatial analysis. There are quite many methods for the analysis of the spatial structure, dynamic models and spatial ontologies. The simplest and oldest measures of the spatial pattern is the Variance:mean ratio method based on the counts of individuals in some kind of sampling units such as quadrats. In the quadrat variance methods, the spatial locations of the sample units are included in the analysis, and the data have to be collected as a complete census in strings or grids of contiguous quadrats [21].

Measures of spatial autocorrelation and autocovariance are derived from the familiar statistical concepts of covariance and correlation [21]. The spatial pattern analysis approach could be used to discover various location patterns. The ArcGIS Spatial Autocorrelation (Moran's I) tool was used to investigate the features of fast food restaurant distribution in Jakarta. The clustered pattern was investigated using Kernel Density Estimation (KDE) to identify the clustered area of fast food restaurants. Moran's I function was applied to find the clustered pattern of values across the study area using a set of fast food restaurant locations [22].

In the Neighbor networks method, the measures of autocorrelation such as Geary's and Moran's can be estimated using not only physical distance, but also for the values, counts, or other measures, at pairs of points that are defined as neighbors by a network of lines joining them [21]. Spectral analysis and related techniques examine periodicity in the spatial pattern of density data by fitting sine and cosine functions to the data and determine which frequencies or wavelengths best fit the data [21]. Wavelet analysis analyzes spatial data related to spectral analysis, which uses a finite template or wavelet rather than sine and cosine functions applied over the length of the data sequence [21].

Fractal dimension describe phenomena that are continuous but not differentiable, so that it seems to have a fractional rather than an integer dimension [21].

There are numerous other techniques, like the Run length and join counts method for one-dimensional data; the Second order point pattern analysis for mapped data for analyzing the mapped positions of objects; the Mark correlation function method for the investigation of the interactions between neighboring trees in a forest. In addition, they also mention the Local Index of Spatial Association method that evaluates how the strength of spatial autocorrelation varies with location within the study area; the Circumcircle method that expands the idea of counting points in circles for completely mapped point data; the Cluster detection method used for detecting clusters of 'diseased' points in a point pattern; the Spatial Analysis by Distance given a number of individuals in each of several quadrats; the and Mantel test for assessing the relationship between two distance matrices, where distance may be of physical location or a measure of some other kind of dissimilarity [21].

Cellular Automata (CA) and Agent-based Model (ABM) are the two prominent dynamic models occupying a large portion of spatial discussions in the last two decades [23]. CA consists of four basic elements: cell, state, transition rules, and neighborhood. The cell represents a spatial shape of CA, the state conveys the possible situation that a cell could have, transition rules determine the changing state of a cell and neighborhood is adjacent cells surrounding the center cell [23]. CA gains considerable attention among geographers and urban planners because urban growth can be easily simulated in CA and it gives intuitive simulation results [23].

ABM can be defined as a combination of three elements: the agent, environment, and interaction. The agent in ABM is anything, which has a discrete entity with a distinct goal, environment is the location where agent performs its tasks and there are two kinds of interaction: agent-to-agents, and agent-to-environment interaction [23]. CA is a concept that suits best to represent the shapes (fabric) of a city; the ribbon, leap frog, natural development, whereas ABM suits best with cases where the interaction of actors involved in urban system are more than one way and actors have complex behavior including learning and adaptation [23].

Ontologies can be very useful in spatial analysis. They can be used for geographic and topographic data sets integration, geospatial data querying, topological querying of multiple map layers and discovering of geographic information services. Ontologies can be constructed for the conceptual dimensions of geographic objects, e.g. for geometry, topology, symbology of representations, and thematic contents [24]. The levels of ontologies can be used to guide processes for the extraction of more general or more detailed information and the use of multiple ontologies allows the extraction of information in different stages of classification [25]. One of the advantages of using an ontology-driven GIS (ODGIS) is the ability of having multiple interpretations to the same geographic feature [25]. Classes are typically defined hierarchically, taking advantage of one of the most important concepts in object-oriented systems: inheritance [25]. In order to represent the diverse character of the geographic entities and avoid the problems of multiple inheritance the objects with roles could be used [25].

5 Conclusions

This paper presents a very basic description of the urban pattern in the form of UML class diagrams covering the aspects of the hierarchy and generalization of patterns and metapatterns. Subsequently, the data model for keeping 3D geometric and topographic data of the urban pattern was reviewed, and the initial solution was chosen. In addition, a short overview of the methods and solutions of spatial analysis, concentrating on the methods based on space syntax, was performed. Some basic components which could be used in further research and computerization of the spatial pattern methodology were extracted:

1. The urban pattern could be constructed using small indivisible spatial elements – metapatterns - that later could be used to identify patterns of different scale and complexity.
2. The context of the urban pattern is mandatory to manage different urban patterns that exist on different hierarchical levels of the city structure.
3. The plain 3DFDS model was sufficient for presenting the main principles, though the research of the advantages of other data models that could be used should be explored in the future.
4. Space syntax method could be used to detect at least some of the urban patterns in spatial data as it provides a set of theories and methods for the analysis of spatial configurations of all kinds and at all scales.
5. Beside space syntax, there are many other methods of spatial structure analysis, like fractal analysis and others, which could be used for the detection and algorithmization of urban patterns.

References

1. Alexander, C.: The Timeless Way of Building. Oxford University Press, New York (1979)
2. Alexander, C.: A Pattern Language. Oxford University Press, New York (1977)
3. Jiang, B., Claramunt, C., Klarqvist, B.: An integration of space syntax into GIS for modelling urban spaces. Int. J. Appl. Earth Obs. Geoinf. **2**, 161–171 (2000). https://doi.org/10.1016/S0303-2434(00)85010-2
4. Matijošaitienė, I., Zaleckis, K., Stankevičė, I.: Sustainable city – a city without crime. Archit. Urban Plan. **7**, 59–64 (2013). https://doi.org/10.7250/aup.2013.007
5. Ahmed, B., Hasan, R., Maniruzzaman, K.M.: Urban morphological change analysis of Dhaka City, Bangladesh, using space syntax. ISPRS Int. J. Geo-Inf. **3**, 1412–1444 (2014). https://doi.org/10.3390/ijgi3041412
6. WP4 Deliverable Report: Urban Pattern Specification. http://www.suburbansolutions.ac.uk/documents/WP4DeliverableReportNov2009.pdf. Accessed 04 July 2018
7. Huisman, O., By, R.A.: Principles of Geographic Information Systems. The International Institute for Geo-Information Science and Earth Observation, Enschede (2009)
8. Billen, R., Zlatanova, S.: 3D spatial relationships model: a useful concept for 3D cadastre? Comput. Environ. Urban Syst. **27**, 411–425 (2003). https://doi.org/10.1016/S0198-9715(02)00040-6
9. Volk, T.: Metapatterns. Columbia University Press, New York (1995)

10. Borgatti, S.P., Eĺerett, M.G.: Models of core/periphery structures. Soc. Netw. **21**, 375–395 (1999). https://doi.org/10.1016/S0378-8733(99)00019-2
11. Major, M.D.: The Syntax of City Space: American Urban Grids. Routledge, London (2018)
12. Dimililer, R., Akyuz, U.: Towards a multi-disciplinary approach in urban design education: art and software (Depthmap) use in urban design of public spaces. EURASIA J. Math. Sci. Technol. Educ. **14**, 1325–1335 (2018). https://doi.org/10.29333/ejmste/81521
13. Qiming, Z., Zhang, W.Z.: A preliminary review on three-dimensional city model. Geo-spat. Inf. Sci. **7**, 79–88 (2012). https://doi.org/10.1007/BF02826641
14. Campagna, M.: GIS for Sustainable Development. CRC Press, Boca Raton (2005)
15. Kyu, S., Ban, J.Y.: Computational algorithms to evaluate design solutions using Space Syntax. Comput. Aided Des. **43**, 664–676 (2011). https://doi.org/10.1016/j.cad.2011.02.011
16. Zlatanova, S., Rahman, A.A., Shi, W.: Topological models and frameworks for 3D spatial objects. Comput. Geosci. **30**, 419–428 (2004). https://doi.org/10.1016/j.cageo.2003.06.004
17. Ariza-Villaverde, A.B., Jiménez-Hornero, F.J., Gutiérrez De Ravé, E.: Multifractal analysis of axial maps applied to the study of urban morphology. Comput. Environ. Urban Syst. **38**, 1–10 (2013). https://doi.org/10.1016/j.compenvurbsys.2012.11.001
18. Depthmap 4 - A Researcher's Handbook. http://www.vr.ucl.ac.uk/depthmap/depthmap4.pdf. Accessed 21 Apr 2018
19. The Space Syntax Toolkit: integrating depthmapX and exploratory spatial analysis workflows in QGIS. https://www.researchgate.net/publication/281856918_The_Space_Syntax_Toolkit_integrating_depthmapX_and_exploratory_spatial_analysis_workflows_in_QGIS. Accessed 04 July 2018
20. Sevtsuk, A., Mekonnen S.M.: Urban network analysis: a new toolbox for ArcGIS. Revue internationale de géomatique **22**, 287–305 (2012). https://doi.org/10.3166/rig.22.287-305
21. Dale, M.R.T., Dixon, P., Fortin, M., Legendre, P., Myers, D.E., Rosenberg, M.S.: Conceptual and mathematical relationships among methods for spatial analysis. Ecography **25**, 558–577 (2002). https://doi.org/10.1034/j.1600-0587.2002.250506.x
22. Widaningrum, D.L., Surjandari, I., Arymurthy, A.M.: Spatial data utilization for location pattern analysis. Procedia Comput. Sci. **124**, 69–76 (2017). https://doi.org/10.1016/j.procs.2017.12.131
23. Wahyudi, A., Liu, Y.: Spatial dynamic models for inclusive cities: a brief concept of Cellular Automata (CA) and Agent-Based Model (ABM). Jurnal Perencanaan Wilayah dan Kota **26**, 54–70 (2015). https://doi.org/10.5614/jpwk.2015.26.1.6
24. Uitermark, H.T., van Oosterom, P.J.M., Mars, N.J.I., Molenaar, M.: Ontology-based geographic data set integration. In: Böhlen, M.H., Jensen, C.S., Scholl, M.O. (eds.) STDBM 1999. LNCS, vol. 1678, pp. 60–78. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-48344-6_4
25. Fonseca, F., Egenhofer, M., Agouris, P., Câmara, G.: Using ontologies for integrated geographic information systems. Trans. GIS **6**, 231–257 (2002). https://doi.org/10.1111/1467-9671.00109

Smart Deployment of Virtual Machines to Reduce Energy Consumption of Cloud Computing Based Data Centers Using Gray Wolf Optimizer

Hossein Shahbazi[1(✉)] and Sepideh Jamshidi-Nejad[2]

[1] Department Information Technology, Mehrastan University, Rasht, Iran
Hossien.shahbazi2000@gmail.com
[2] Young Researchers and Elite Club, Tehran_Shomal Branch, Islamic Azad University, Tehran, Iran
jamshidi1270@gmail.com

Abstract. The growth in demand for using cloud computing resources at massive data centers has led to high consumption of energy and, consequently, increased operating costs. Integration of cloud resources makes it possible to save time on the migration of loaded and unprocessed data centers, to qualified data centers, the release of idle nodes, and the reduction of virtual machine virtualization migration.

One of the most important challenges is to choose the method of embedding virtual machines that are migrating to the node. Therefore, in this paper, a solution is proposed to reduce energy consumption in cloud data centers. In this solution, the gray wolf optimizer is used to properly assign the virtual machine to the appropriate node. The methodology was simulated with the Claudios software. The results of the simulation indicate a decrease in the number of virtual machines migrating, increasing the efficiency of migration and reducing energy consumption.

Keywords: Energy consumption reduction · Cloud computing
Gray wolf optimizer · Cloud migration

1 Introduction

Many researchers put their work on scheduling or allocating resources, work, and so on. In cloud computing, scheduling is a critical issue because the cloud provider has to serve multiple users. The main purpose of the scheduling algorithm is to minimize runtime and cost.

The allocation of resources is also a critical issue because in the cloud every job must be mapped to run on a physical server. This mapping is possible in a variety of ways, but it is difficult to find a suitable server to host the work and to ensure that it delivers time, cost-effectiveness and efficiency. This is quite the same as the multidimensional backpack or bin packing, with the difference that it is considered in the allocation of resources.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 164–177, 2018.
https://doi.org/10.1007/978-3-319-99972-2_13

The term cloud is a metaphorical concept that refers to the huge repository of hardware and software that can be easily accessed through the Internet. Data centers are among those high processing power centers that can respond to thousands of users' computer needs. Most of these data centers utilize virtualization techniques, and in this way provide a distributed computing environment in which cloud services are provided on a virtual machine platform and virtual machines run on the physical host platform.

As mentioned, one of the most important problems with data centers is their high energy consumption, which is neglected following the competition of large companies for the rapid development of data centers. One way to reduce energy consumption in data centers is to reduce the processing burden of hosts, but it should be noted that a host, even in idle mode, will have only 30% less energy than the full processor power [2]. In the first step, the best way to reduce energy consumption is to optimize the virtual machines in the physical hosts so that the least possible number of these hosts is used and in the second stage it is important to balance the processing load of these hosts.

The issue can be examined in two respects: the first aspect is the acceptance of new requests and the allocation of virtual machines for them and the placement of virtual machines on the hosts. This is a type of grouping called bin packing grouping which is divided into two categories: Type I: A multidimensional bin packing problem that can be changed in the final solution to this kind of problem by how items can be folded into boxes.

The second type is the multi-capacity bin packing in which the final solution is not affected by the way that items can be folded into boxes. Common algorithms such as meta-heuristic algorithms are used to solve bin packing problems and to achieve optimal response. In order to achieve the optimal answer, it is necessary to reduce the number of physical servers in hosts, which would result in a significant reduction in energy consumption like meta-heuristic algorithms. In this research Wolfgang optimizer is used to solve the bin packing problem.

The second aspect is the optimization of the virtual machines on the hosts during processing of the system, which in turn leads to the lowest energy consumption in the data centers. By using virtual machine migration techniques, this may be the case that there is no need to physically transfer the virtual machine file through the network, and only the main contents of the virtual machine's memory are transmitted through the network to the destination host in the data centers.

2 Literature Review

In recent years, many studies have been conducted on the problem of optimizing resource allocation in order to reduce fuel consumption. Martin Bichler et al. (2009) have implemented a linear programming approach to assign virtual machines. Although they used binary decision variables, they only responded to static source allocation and did not cover the dynamic nature of the cloud environment.

Babroff et al. (2011) presented an algorithmic work flow for the integration of physical machines, based on a periodic minimization of clear cars and a breach of service quality warranties. Spitcomp and colleagues (2012) presented a linear formula for the dynamic integration of virtual machines and mapped some of the virtual machines to servers with specific features.

Also, for the number of migrations, thresholds were considered that would not exceed this limit. They also paid attention to the cost criterion in their formulation [19]. Among other researches in this field is the research by Essental et al. (2013). They have two types of hard SLA and Soft SLA performance levels for their services. In the hard type, if the service provided is not exactly at the requested level and not in the contract, then no revenue will come to the server. But in the kind of Soft, customers can withstand even weaker performance levels. But the income earned is decreasing according to the pricing table [20]. In order to solve the resource allocation problem, in addition to the 2014 power factor, an algorithm was proposed by Spitcomp and colleagues. In this algorithm, the profit variable was also considered. In each virtual machine assignment to the host, the resulting profit was calculated. Then, the best place for each virtual machine was considered with the changes that were first fitted in the base algorithm.

Among the researches on the virtual machine migration, we can mention the research by Blogglaso et al. (2012). In this research, the minimum migration policy for virtual machines migration and the FF algorithm were used for initial deployment of virtual machines. In the FF algorithm, the smaller the number of virtual machines and hosts, the best results are obtained and vice versa.

Banergy et al. have used the Locations-aware Dynamic Resource Allocation Model for cloud environments. This model has two important tasks. First, the decision to determine the placement of virtual machines and the second to decide on their migration. In this way, if the provider receives a request from the user, it finds the provider of the closest physical host in the area and assigns it after the appropriate level of productivity with a particular physical machine. The advantage of this model is that it does not reduce the functionality of the data center, but the disadvantage of this model is that it is not enforceable in the real environment of cloud computing.

A number of studies have used the Modified Best Fit Decreasing (MBFD) algorithm. This algorithm is a retrieval method based on determining a threshold value for the efficiency of the Single Threshold and Double Threshold for hosts. So that virtual machines are placed on hosts so that host productivity does not exceed the threshold value. At each time period, all virtual machines are reallocated using the MBFD algorithm. If the capacity of a host is below the limit, all virtual machines on which they are located should be migrated to reduce energy consumption and lower costs.

In addition to the lower limit mentioned above, a high limit is also considered for the productivity of the hosts, so if the amount of use of the host capacity exceeds its maximum, a number of virtual machines should migrate, so it is sometimes possible We have overflow in host. In another study, Optimal VM algorithm was used. Using this algorithm, the energy consumption of virtual machines was calculated on all physical machines and finally placed on the host that had the lowest energy consumption.

The time complexity of this algorithm is O (m ^ n), which expands exponentially [6]. The Best Fit Random Heuristic algorithm is also described in [7]. This algorithm randomly selects and allocates a thousand different numbers that meet the needs of virtual machines that do not spoil the capacity of physical hosts, and calculates the energy consumption of these thousand allocations. Finally, it performs the lowest energy usage. The complexity of this algorithm is exquisite, and therefore, the algorithm is not optimal. Greedy Heuristic algorithm is reviewed in [6, 8]. Based on this algorithm, all virtual machines are first sorted according to their sorting factor, and then all possible allocations in which the conditions are not violated are calculated. Assignment that has the lowest energy consumption is done. The time complexity of this algorithm is not optimal.

In [8], the Modified First Fit Heuristic algorithm is also used, which arranges both virtual machines and servers based on the ordered factor. Then, for all virtual machines, the list of physical hosts is searched. The first physical host that has enough space for the virtual machine is assigned. Since the two lists are arranged according to the factors, the first one found is also the most appropriate. The time complexity of this algorithm is equal to O (nlogn) O (mlogm + n), and in the worst case, it is calculated as O (n.m). In a number of studies, a group genetic algorithm has been used for the initial deployment of virtual machines and the least migration policy for the virtual machines migration. The result obtained from the implementation of this algorithm in comparison with the first best algorithm, shows a better performance [9, 10].

Mesbah Abdul Qadir et al. (2016) used the gray wolf optimizer to optimize the timing of the unit's power storage unit in smart homes. Their goal was to reduce the cost of power consumption and load balances in the electricity grid. The algorithm determines how much power is taken from the network and uses power from the stored unit for every hour of the day. The algorithm was tested on data collected from the Chicago area by the US Department of Energy. The results were compared with particle optimization in the same data set. Gray wolf optimizer saves up to 25.57% in cost compared to particle swarm optimization algorithm [16].

Zing San and colleagues (2016) provided a comprehensive survey of the most research activities related to resource management of data centers in order to optimize the use of resources. They first described the problem of over provisioning resources in data centers. Then, they summarize two important components in the resource management platform and use the benefits of job resource prediction precision in resource management. Subsequently, the management of resources in a data center was divided into three categories: 1. Virtual machine-based, 2. Physical machine-based, 3. Utility resource management mechanisms. This paper discusses the performance degradation of these three types of resource management in an inhomogeneous data center. This essentially provides a timely survey on resource management in a data center and provides a comprehensive reference for further research in this area.

In [22] checked further analyzed energy consumption management questions of the virtual cloud computing platform. It eventually gives people a clearer understanding of energy consumption management of virtual cloud computing platform and brings more help to various aspects of people's live, work and so on.

It is explained in [23] that Consolidation VM can dynamically react to the increasing demand for resources to reallocate the VM back when necessary to avoid degradation in the performance of cloud computing system as a whole. VM consolidation procedures that were implemented into OpenStack-Neat can be divided into several components, namely the data collector, the local manager and global manager.

The components used to handle the consolidation procedure is to decide when the host is considered as host underload and overload, the selection of VM to migrate from overload host and placement of the selected VM on the selected host. Value CPU Time used as fill to get the average value in MHz CPU utilization within a specific time period. The average value of a VM's CPU utilization in getting from the current CPU_TIME is reduced by CPU_TIME from the previous data retrieval multiplied by the maximum frequency of the CPU. The calculation result is divided by the time making CPU_TIME when it is reduced to the time taking previous CPU_TIME multiplied by milliseconds.

3 Physical Resource Management

An important challenge facing cloud service providers is the effective management of physical resources hosted by physical infrastructure. It seems that the initial placement of workflows in which each virtual machine has a primary share of the resources is not compatible with the dynamic nature of the workflow of the cloud environments, which can at any given moment face with the decrease or increase of their demand. Therefore, important points arise in data centers where the resources required by virtual machines will be more than the capacity of the data center hosted on them.

Therefore, on a large scale, these resources require automated management to address the problems of these areas. One of the most important tasks of cloud service providers is the management of resources. Cloud consumers send cloud service requests to the cloud from anywhere in the world. Cloud service providers should assure customers that their needs are fully met. Until recently, achieving high performance without considering the cost of increasing energy was only a concern when allocating resources [10].

Energy consumption in cloud environments is examined from two perspectives: The first perspective, static power management, which is more related to hardware, is beyond our scope of work and the second view is the dynamic management of energy consumption, which is the main topic of our discussion [14]. The process of collecting services running on several physical machines and executing them on a smaller number of sources, so that the level of performance requested by the user is not violated, is called integrity, and allows service providers use as much as possible to have fewer physical resources for Providing users with their needs, and as a result of lower energy consumption, the overall cost of service will be reduced.

Integrity action can lead to performance failure if not done correctly. If resources are not provided, we will be faced with an increase in response time, which can lead to error and contract breach with the user. Resource integrity can be described in four stages of monitoring, estimating, reorganizing, and commissioning. The first step is to monitor the resources used by virtual machines. In the estimation phase, according to the data collected in the monitoring phase, the need for virtual machines is known at any moment, and if a machine can not obtain its own resource or exceed its productivity by the specified level or a physical machine that is lacking resources (such as a processor, memory, bandwidth, etc.) is known to serve hosted virtual machines.

So at this point, the focal host, which is the same machine that exceeds threshold, will be found. The purpose of transferring virtual machines from this host to another is through live migration, so that the contract and user needs are not violated [10].

This kind of migration allows the system to resume its work from the point where the system was interrupted after the system has migrated without initiating it from the beginning. It has been shown that system disruption is very low due to this kind of migration (i.e., 06 ms). As a result, performance declines are not much visible. The reorganization is the stage in which the selected virtual machines are mapped to a new host to continue their execution.

4 Gray Wolf Optimizer

The gray wolf optimizer was introduced by Mirjalili et al. [1]. Gray wolf is considered to be the highest level of hunters because there is no natural hunter for this species of animals. Gray wolves are usually grouped in groups of 5 to 15. Leaders who are known as alpha are responsible for hunting, resting, moving time, and other issues.

Interestingly, Alpha wolves are not necessarily the strongest wolf in the band, but they are best-wolf in terms of group management. This suggests that the discipline of the group is much more important than power. The second group of gray wolf hierarchies belongs to the beta category. Beta wolves help alpha wolves in decision making and other group activities. When alpha wolves are aged or dead, beta wolves are the best alternative to them. The lowest rank of the gray wolf hierarchy is the Omega wolf, which inevitably needs to be added to the higher categories. Wolves that are not in any of the alpha, beta or omega categories are delta. Delta wolves follow the alpha and beta categories but dominate the omega wolves. Important members of the group such as scouts, watchers, hunters belong to this category.

In addition to the series of social life, another social behavior of wolves is their group hunting. The most important hunting phases of gray wolves are as follows:

1. Going on, chasing and catching the prey
2. Chasing, blocking and harassing the prey until it stops moving
3. Attack on the prey.

4.1 Gray Optimizer Modeling

Generally in the gray wolf optimizer, the search process is created by creating a random population of gray wolves (candidate solutions). During the repetition, the alpha, beta and delta wolves estimate the probable position of the prey. Then update all available solutions and distance from the prey. The parameter a (distance between candidate and prey) reduces from 2 to 0 to emphasize local and global searches. Candidate solutions, if $|A| > 1$, tend to get away from the prey and tend to prey for $|A| < 1$.

Local and global search is highly dependent on how parameter A (proposed algorithm parameter) functions and how elitism is selected to update population of candidate solutions. This approach ultimately leads to the close of α, β, and δ, thereby reducing local and global searches for global optimization. Reducing local and global search will trigger gray wolf optimizer to stay in the optimal local trap. In this paper, two techniques have been used to improve local and global searches and the transition from local optimization to global optimization.

5 Methodology

The best solution to reduce energy consumption in data centers is to optimize the virtual machines in physical hosts, so that the least possible number of these hosts is used. In previous studies, more emphasis has been placed on the lesser intelligence and lack of collective intelligence methods, while in this study, the gray wolf intelligence optimizer was used to manage host machines. Also, the initial deployment of virtual machines in hosts is important before starting processing in this study.

Note that until now, the gray wolf optimizer has not been used to solve hard bin packing optimization problem. Therefore, in this paper we seek to solve the multi-capacity bin packing problem in the form of energy reduction in cloud computing data centers using this algorithm. The following is defined:

In the issue of the deployment of virtual machines, virtual machines, items and physical hosts, boxes were considered. Because physical hosts have different memory resources, processing power and bandwidth, and each virtual machine needs these resources, the purpose of this study is to allocate virtual machines to the hosts, so that the least number of physical hosts are used, so here the gray wolf optimizer was used to allocate optimal virtual machines to the host.

The routine of this algorithm is that in the first step, each wolf, which is in fact a solution to the problem, is randomly initialized with respect to the following relationships. The fitting value of each a and C, A in the second stage of the vectors of the search agent coefficients (wolf) is calculated for the third stage (see Fig. 1), among them, the first best answer is chosen as wolf alpha, the second best answer is chosen as wolf beta, and the third The best answer is chosen as wolf delta. (Steps 4 to 6). In each replica, the wolves' positions are updated according to the location of the three wolf alpha, beta and delta wolves. Also, the values of the vectors of the coefficients a, A and C are updated. Given the new situation, the value of the fitting function of the wolves is calculated, among which alpha, beta and delta wolves will be selected.

5.1 Optimal Gray Wolf Optimizer Modeling

In this section, the mathematical model of social behavior and gray wolves hunting is presented to introduce this algorithm, based on the description of how gray wolf's social life and hunting is. In order to model the social behavior of wolves, we generate a random population of solutions and introduce the first optimal solution called alpha (α), the second and third optimal solutions, respectively, called beta (β) and delta (δ). Other solutions are considered as omega group wolves (ω). The gray wolf optimizer uses three α, β, and δ solutions to guide hunting (optimization), and ω answers follow these three. In order to model the three phases mentioned previously, it is necessary first to determine the points around the prey. Then he moves to the prey and eventually attacks the prey. To determine the points around the prey, we use Eqs. 1 and 2:

$$\vec{D} = \left| \vec{C}.\overrightarrow{X_P}(t) - \vec{X}(t) \right| \tag{1}$$

$$\vec{X}(t) = \overrightarrow{X_P}(t) - \vec{A}.\vec{D} \tag{2}$$

```
Modified Grey Wolf Selection Policy Algorithm
Initialize the grey wolf population Xi (i = 1, 2, ..., n)
Initialize a, A, and C
Calculate the fitness of each search agent
Xα=the best search agent
Xβ=the second best search agent
Xδ=the third best search agent
while (t < Max number of iterations)
  for each search agent
   Update the position of the current
   search agent by equation (7)
  end for
 Update a, A, and C
 Calculate the fitness of all search agents
 Update Xα, Xβ, and Xδ
 t=t+1
end while
return Xα
return optimize VM and host mapping
```

Fig. 1. The proposed algorithm for reducing energy consumption of cloud computing data centers using the gray wolf optimizer.

In the Eqs. (1) and (2), t is the current repetition, A and C are coefficient vectors, (X_P) is the position of the prey and X is the position vector of a gray wolf. The vectors A and C are also calculated with Eqs. 3 and 4:

$$\vec{A} = 2\vec{a}.\vec{r_1} - \vec{a} \tag{3}$$

$$\vec{C} = 2\vec{r_2} \tag{4}$$

Note that the components $\vec{a}$ decrease linearly during the repetition from 2 to 0, and r_2 and r_1 are random vectors in the interval [1 and 0].

In order to mathematical modeling of the gray wolf hunting behavior, given that in the initial search space there is no idea about the position of the prey, it is assumed that α (the best candidate for resolution), β and δ are the best knowledge of the position of the prey. So, keep the three best answers we have so far and force other factors such as ω to update our position. Formulas 5, 6 and 7 show the mathematical modeling of hunting and approaching the prey (optimal value):

$$\vec{D_j} = \left|\vec{C_i}.\vec{X_j} - \vec{X}\right|, \vec{X_i} = \vec{X_j} - \vec{A_i}.\left(\vec{D_j}\right) \tag{5}$$

$$(i,j) \in \{(1,\alpha),(2,\beta),(3,\delta)\} \tag{6}$$

After calculating Xi, the point X is updated with the following formula:

$$\vec{X}(t+1) = \frac{\overrightarrow{X_1} + \overrightarrow{X_2} + \overrightarrow{X_3}}{3} \tag{7}$$

The final position will be defined in a random location located in a circle defined by the positions α, β and δ in the search space. In other words, α, β, and δ estimate the position of the prey, and other wolves (existing solutions) update their position randomly around the prey or optimal point.

6 Simulation

To determine the resource management issue, many parameters have been identified. We briefly discuss the most important of these parameters. To simulate our resource management issue, we used a CoreTMi5 computer with 4 GB of original memory. The real clock of 2.66 GHz processor was set and the operating system on which the simulator was running (Home Premium 64-bit) was chosen. It should be noted that both virtual machines and hosts (physical machines) are considered homogeneous, and without losing the whole problem, since the highest energy consumption is among the resources belonging to the processor [11], we just consider that in our experiments, so we deal with a single-dimensional bin packing problem in this research.

To simulate 800 hosts, 1052 virtual machines and 1052 units of work has been built. Each task was mapped onto a unique virtual machine. We also compare the proposed method with three algorithms called FF, genetics, and Round-Robin scheduling (RR). To equate the comparison of these methods with each other in simulation for all, an end time of 24 * 360 was considered, which in fact represents the seconds of length of a day and all algorithms at this time were performed. As soon as the clock simulator arrives to this special number, the performance will stop. Obviously, this number is not exactly the same as the actual seconds. The LANETLAB

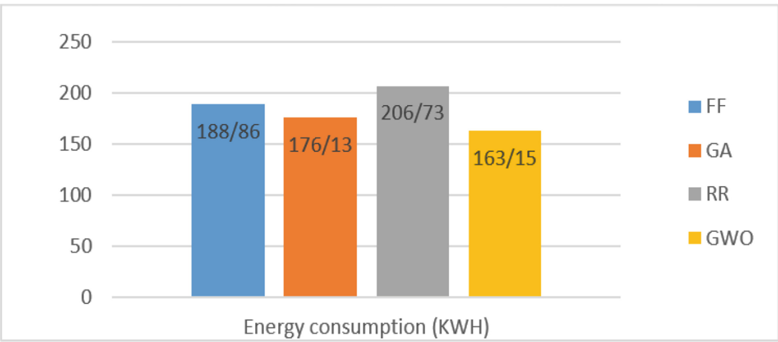


Fig. 2. Energy consumption (KWH)

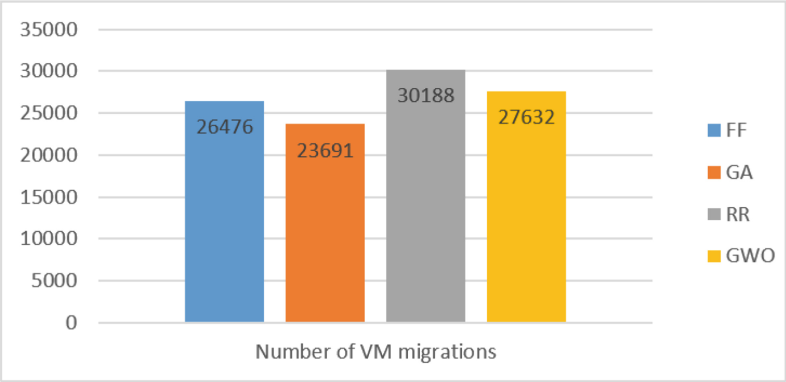


Fig. 3. Number of VM migrations

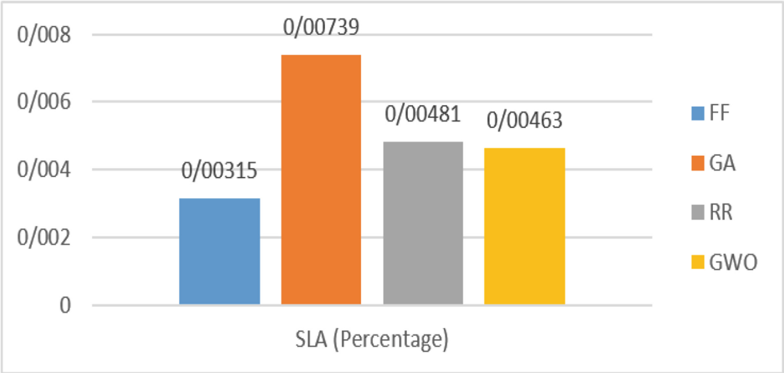


Fig. 4. SLA (percentage)

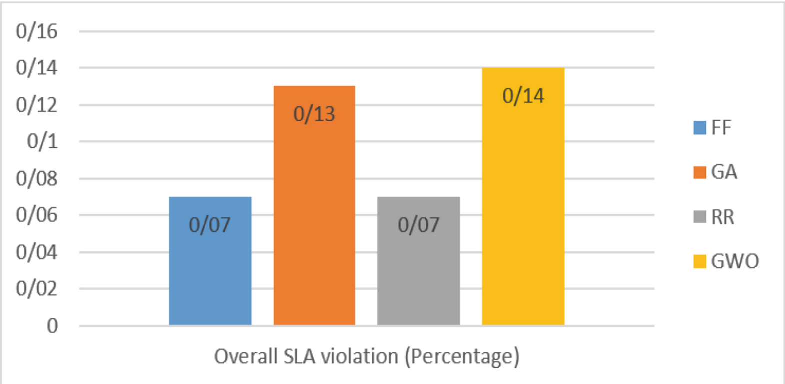


Fig. 5. Overall SLA violation (percentage)

2011 data set was used to evaluate the system. To implement the proposed method, one of the most well-known cloud simulation environments, CloudSim, was used. The proposed algorithm is presented in Fig. 1.

In Figs. 2, 3, 4, 5, 6, 7, 8 and 9 results of the implementation of the proposed algorithm are given. Note that in all graphs the rightmost columns, GWO, represents the proposed method of this research. In Fig. 2, the energy consumption is visible, with the lowest consumption associated with the GWO algorithm. In Fig. 3, the number of virtual machine migrations is shown in three other algorithms, with GWO ranking third in the number of migrations.

In service level agreement, GWO is better than the two RR and GA algorithms. This is shown in Fig. 4. The violations of the global and medium-level agreement are depicted in Figs. 5 and 6, which are more visible in the average GWO. In Figs. 7, 8 and

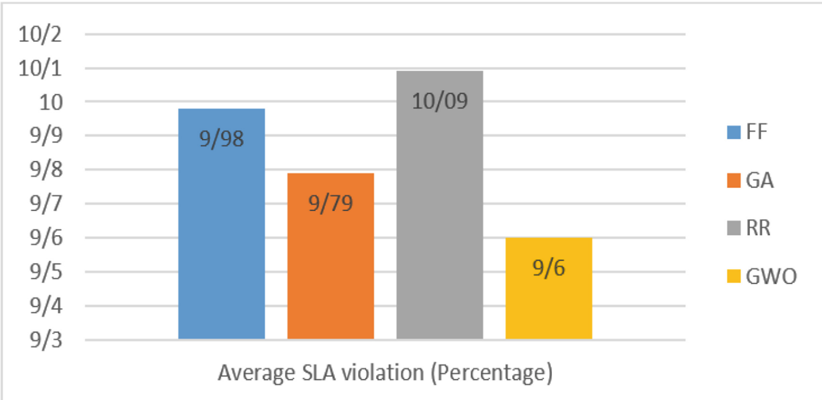


Fig. 6. Average SLA violation (percentage)

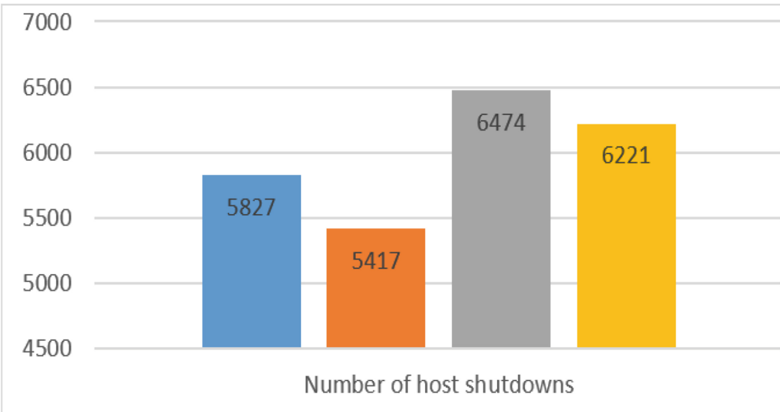


Fig. 7. Number of host shutdowns

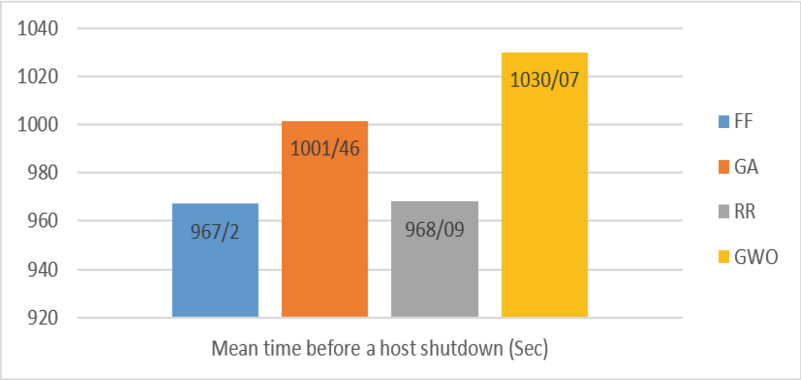


Fig. 8. Mean time before a host shutdown (Sec)

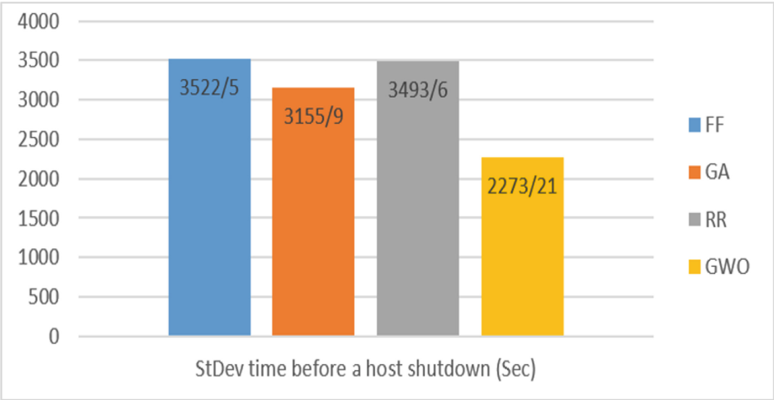


Fig. 9. STDEV time before a host shutdown (Sec)

9, the number of offsets hosts, mean time and standard deviation of the proposed algorithm based on GWO is compared with three algorithms, in all cases the proposed algorithm is better than the other three algorithms.

7 Conclusions and Future Researches

The best solution to reduce energy consumption in data centers is to optimize the virtual machines in physical hosts, so that the least possible number of these hosts is used, because in past researches, the main focus was based on lesser intelligence methods that lack the properties of collective intelligence, this paper focuses on the initial deployment of virtual machines in hosts prior to processing with intelligent gray wolf optimizer. So far, the gray wolf optimizer has not been used to solve the bin packing problem, but here the problem of multi-capacity bin packing in the form of energy reduction in cloud computing data centers has been addressed.

In the issue of deploying virtual machines, they can be considered as physical boxes and items and our goal is to allocate virtual machines to hosts so that the least number of physical hosts are used. Therefore, the gray wolf optimizer was used to allocate optimal virtual machines to the host. The energy consumption of the gray wolf optimizer has been noticeably improved compared to other methods. As a result, fewer servers at the beginning of the work can be used; in other words, there are fewer servers at the start of work than other methods. The average time that physical servers are turned off is higher in the proposed method than in other algorithms.

As more virtual servers are on a host, the timeout for these virtual machines is also increased, resulting in the increase of a complete physical shutdown time of a host. The total number of offset hosts will have a huge impact on overall energy consumption, and the greater the total number of offset hosts, the total amount of energy consumed will be reduced, but this component can not be an appropriate component alone, but should be compared to the average off time of servers. The average server shutdown time in the RR method is lower, which indicates that virtual machines are spread over more servers. Also, our proposed method has a higher service level agreement violation rate than its competitors. Perhaps the reason for such an outcome is due to the accumulation of greater load.

To improve the proposed method, much more can be done. For example, using a known temperature method, it is possible to simulate and measure the amount of heat generated in different parts of the system and provide an algorithm that presents the temperature of the environment and servers before inserting as an important factor to consider. In this research, it is assumed that virtual machines are separated from each other. Another future work could be to address the issue of communication and traffic between virtual machines.

References

1. Ebrahimi, F., Farahi, A., Farhoudinejad, A.: Review of resource allocation methods and their importance in the computer environment (2012)
2. Farhadi, A., Varjani, A.V.: The smart deployment of virtual machines in cloud computing based data centers using a group genetic algorithm. In: The 11th Conference on Intelligent Systems of Iran, March 2012
3. Beloglazov, A., Abawajy, J., Buyy, R.: Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. Future Gener. Comput. Syst. **28**, 755–768 (2012). Computer Systems 27, 1028-10301, 2011
4. Banerjee, A., Mukherjee, T., Varsamopoulos, G., Gupta, S.K.: Integrating cooling awareness with thermal aware workload placement for HPC data centers. Sustain. Comput. Inform. Syst. **1**(2), 134–150 (2011)
5. Barroso, L.A., Olzle, U.H.: The case for energy-proportional computing. IEEE Comput. **14** (12), 33–37 (2007)
6. Chung, S., Tam, H.K., Tam, L.M., Zhang, T.: A new optimization method, the algorithm of changes, for bin packing problem, pp. 994–999. IEEE (2010). 978-1-4244-6439-5/10

7. Tam, S.C., Tam, H.K., Tam, L.M., Zhang, T.: A new optimization method, the algorithm of changes, for bin packing problem, vol. 15, pp. 864–877 (2010)
8. Krishnadhan, D.: Extension of cloudsim: cloud computing simulator. s.l.: A Thesis Submitted in partial fulfillment of the requirements for the degree of Master of Technology under the guidance of Prof. Purushottam Kulkarni and Prof. UmeshBellur (2013)
9. Ferreto, T.C., Netto, M.A., Calheiros, R.N., De Rose, C.A.: Server consolidation with migration control for virtualized data centers. Future Gener. Comput. Syst. **27**(8), 1027–1034 (2011)
10. Shaw, P.: A constraint for bin packing. In: Wallace, M. (ed.) CP 2004. LNCS, vol. 3258, pp. 648–662. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30201-8_47
11. Li, X., Qian, Z., Lu, S., Wu, J.: Energy efficient virtual machine placement algorithm with balanced and improved resource utilization in a data center. Math. Comput. Model. **58**(5), 1222–1235 (2013)
12. Orgerie, A.C., de Assuncao, M.D., Lefevre, L.: A survey on techniques for improving the energy efficiency of large scale distributed systems. ACM Comput. Surv. (CSUR), **46**(4) (2014)
13. Tayal, S.: Task scheduling optimization for cloud computing system. Int. J. Adv. Eng. Sci. Technol. **5**(2), 11–15 (2011)
14. Tiago, C.F., Marco, A.N., Rodrigo, N.C., César, A.D.R.: Server consolidation with migration control for virtualized data centers. Future Gener. Comput. Syst. **27**(8), 1027–1034 (2011)
15. Vijindra, Shenai, S.: Survey of scheduling issues in cloud computing. Procedia Eng. **38**, 2881–2888 (2012)
16. Abdulgader, M., Lakshminarayanan, S., Kaur, D.: Efficient energy management for smart homes with grey wolf optimizer (2017)
17. Sun, X., Ansari, N., Wang, R.: Optimizing resource utilization of a data center. IEEE Commun. Surv. Tutor. **18**(4), 2822 (2016)
18. Speitkamp, B., Bichler, M.: A mathematical programming approach for server consolidation (2010)
19. Bobroff, N., Kochut, A., Beaty, K.A.: Dynamic placement of virtual machines for managing sla violations. In: The 10th IFIP/IEEE International Symposium on Integrated Network Management, IM 2007, pp. 119–128 (2007)
20. Esnault, A., Feller, E., Morin, C.: Energy-aware distributed ant colony based virtual machine consolidation in IaaS clouds, dumas-00725215, version 1–24 Aug, 4–6 (2012)
21. Xu, L., Wang, W., Zhang, X.: Oriented-SLA and energy-efficient virtual machine management strategy of cloud data centers. Int. J. Grid Distrib. Comput. **9**(1), 237–248 (2016). http://dx.doi.org/10.14257/ijgdc.2016.9.1.24
22. Li, L.: Energy consumption management of virtual cloud computing platform. In: 2017 IOP Conference Series Earth and Environmental Science, vol. 94, p. 012193 (2017)
23. Fauzi, A., Mulyadi, E., Fadil, A., Idhom, M.: Management of virtual machine as an energy conservation in private cloud computing system. In: MATEC Web Conferences, vol. 58, p. 03008 (2016). https://doi.org/10.1051/mateconf/20165803008

Problem Domain Knowledge Driven Generation of UML Models

Ilona Veitaite(✉) and Audrius Lopata

Kaunas Faculty, Institute of Applied Informatics, Vilnius University, Kaunas, Lithuania
{Ilona.Veitaite,Audrius.Lopata}@knf.vu.lt

Abstract. The main scope of the article is to present how the quality of stored problem domain information in Enterprise model (EM) is significant and important in Unified Modelling Language (UML) models generation process from Enterprise model. The generation process is explained by top-level transformation algorithm, which is presented in details and depicted by using algorithm's step by step description. The importance of information quality and fullness is represented by the example of Business Rule element's, which is stored in Enterprise model, significance for different UML models.

Keywords: UML · Unified Modelling Language · Enterprise model Transformation algorithm · Knowledge-based IS engineering

1 Introduction

There have been and still are pretty many efforts for the analysis of UML models generation from different knowledge-based models, merging and uniting different modelling languages, frameworks and patterns and even generation from natural language specifications [2–4, 9].

The Enterprise meta-model (EMM) is considered to be the significant conceptual structure for problem domain knowledge acquisition for the aims of IS development. This meta-model handles Enterprise model structure, and therefore, Enterprise model stores knowledge that is needed for whole IS development process and can be used in all IS development life cycle phases [5–8].

In recent years UML models are having a rising regard from. It is a very daring objective for analysis of UML models since the knowledge about an enterprise system is distributed within several model approaches. UML models are preserved to decrease the confusion of the problem with the increase of enterprise turnovers. By operating UML models, knowledge from Enterprise model can be efficiently represented and can be used in all phases of IS development life cycle [10, 11, 13].

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 178–186, 2018.
https://doi.org/10.1007/978-3-319-99972-2_14

2 Generation of UML Models Using Enterprise Meta-model

Enterprise meta-model is formally defined enterprise model structure, which consists of a formalized Enterprise model in line with the general principles of control theory (Fig. 1). Enterprise model is the main source of the necessary knowledge of the particular problem domain for IS engineering and IS re-engineering processes [6, 7].

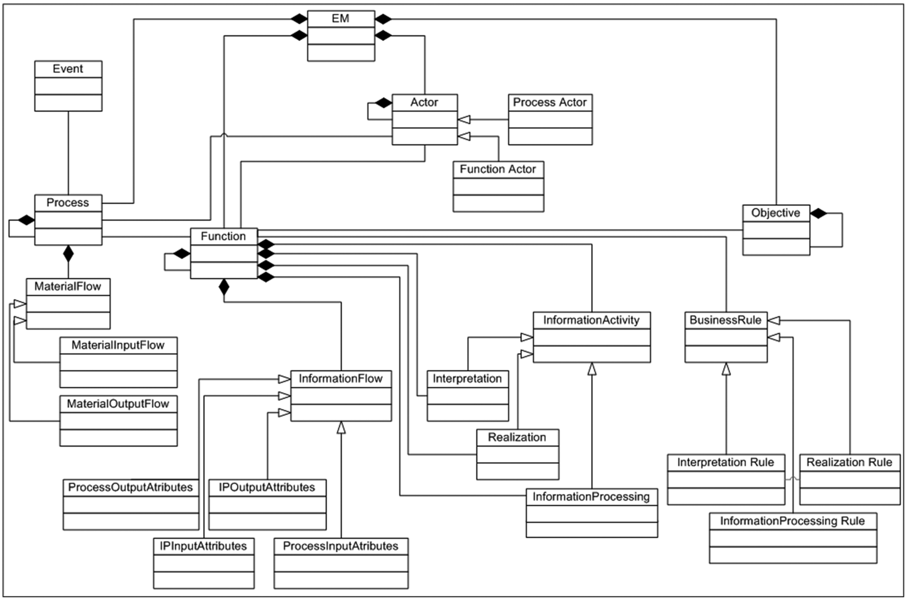


Fig. 1. Enterprise meta-model class diagram [6, 7]

All UML models: static and dynamic can be generated using Enterprise model transformation algorithms. Particular UML model must be selected for generation process, after this selection, the initial – starting element of this particular UML model must be identified from Enterprise model. Therefore, all related elements must be identified according the initial element and all these related elements must be linked regarding constraints i.e. business rules, obligatory for particular UML model type which was selected in the beginning [1, 12].

This kind of system definition is quite tangled, because most of the information from the Enterprise model overlays in the UML models and expresses the same matters just in different approaches, as it is explained in the next chapter. Therefore identification of particular UML model for generation process has high significance, because of regarding this selection relies generated element value for system development process [1, 16].

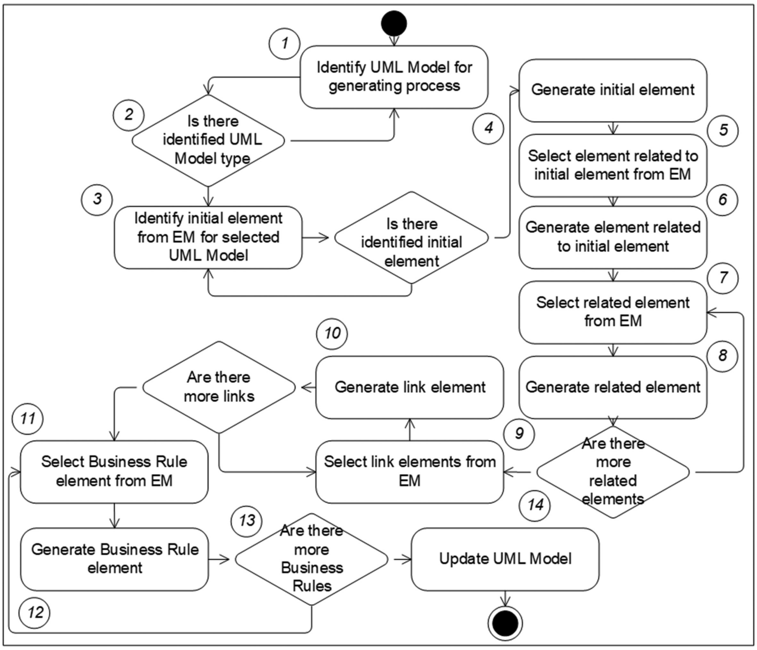


Fig. 2. The top level algorithm of UML models generation using normalised EMM

The algorithm of UML model generation using Enterprise model is presented in the Fig. 2:

- Step 1: Particular UML model for generation from Enterprise model process is identified and selected.
- Step 2: If the particular UML model for generation from Enterprise model process is selected then algorithm process is continued, else the particular UML model for generation from Enterprise model process must be selected.
- Step 3: First element from Enterprise model is selected for UML model, identified previously, generation process.
- Step 4: If the selected Enterprise model element is initial UML model element, then initial element is generated, else the other Enterprise model element must be selected (the selected element must be initial element).
- Step 5: The element related to the initial element is selected from Enterprise model.
- Step 6: The element related to the initial element is generated as UML model element.
- Step 7: The element related to the previous element is selected from Enterprise model.
- Step 8: The element related to the previous element is generated as UML model element.

- Step 9: If there are more related elements, then they are selected from Enterprise model and generated as UML model elements one by one, else the link element is selected from Enterprise model.
- Step 10: The link element is generated as UML model element.
- Step 11: If there are more links, then they are selected from Enterprise model and generated as UML model elements one by one, else the Business Rule element is selected from Enterprise model.
- Step 12: The Business Rule element is generated as UML model element.
- Step 13: If there are more Business Rules, then they are selected from Enterprise model and generated as UML model elements one by one, else the generated UML model is updated with all elements, links and constraints.
- Step 14: Generation process is finished.

3 Business Rule Element's Variations

Design methods of information systems refers the settlement of systems engineering actions, i.e. in what manner, how, and which UML model to use in the information system development process and how to fulfil the process. Many of them are based on different types of models depicting varying aspects of the system properties. Significance of every UML model element can be defined particularly, but more important is the fact that every model is the projection of the system (Fig. 3). Business Rule element from Enterprise model can indicate different view of the system in different UML model.

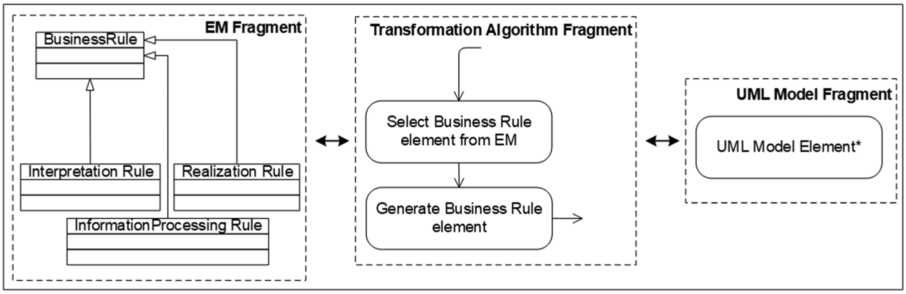


Fig. 3. Business Rule element in EM and in UML model. The transformation algorithm's 11 and 12 steps

Below there are presented generated UML model elements after transformation algorithm's 11 and 12 steps: Business Rule element generation from Enterprise model in UML dynamic part models.

In case of UML Use Case models, which describe a series of actions that some system or systems should or can implement in contribution with one or more external users of the system [15], from EM Business Rule elements can be generated three UML Use Case elements (Table 1): Extend, Include, Association [14, 15].

Table 1. UML Use Case model elements [14, 15]

EM element	UML Use Case model element	Description
Business Rule	Extend	Extend is a directed relationship that specifies how and when the behaviour defined in usually supplementary (optional) extending use case can be inserted into the behaviour defined in the extended use case
	Include	Use case include is a directed relationship between two use cases which is used to show that behaviour of the included use case is inserted into the behaviour of the including use case
	Association	Each use case represents a unit of useful functionality that subjects provide to actors. An association between an actor and a use case indicates that the actor and the use case somehow interact or communicate with each other

Table 2. UML Activity model elements [14, 15]

EM element	UML Activity model element	Description
Business Rules	Control Nodes	Used to coordinate the flows between other nodes. It includes: initial, flow final, activity final, decision, merge, fork, join

In case of UML Activity model, which shows flow of control or object flow with emphasis on the sequence and conditions of the flow [15], from EM Business Rule elements can be generated all types of UML Activity Control Nodes elements (Table 2).

In case of UML State machine model, which is used for modelling discrete behaviour through finite state transitions [15], from EM Business Rule elements can be generated one UML State machine model element: Pseudostate (Table 3), and also state machines can be used to express the usage protocol of part of a system as UML Protocol State machine, and in this case, from EM Business Rule elements can be generated one UML Protocol State machine model element: Protocol Transition (Table 4).

Table 3. UML State Machine model elements [14, 15]

EM element	UML State Machine model element	Description
Business Rule	Pseudostate	An abstract node that encompasses different types of transient vertices in the state machine graph

Table 4. UML Protocol State Machine model elements [14, 15]

EM element	UML State Machine model element	Description
Business Rule	Protocol Transition	Used for the protocol state machines which specifies a legal transition for an operation

In case of UML Sequence model, which focuses on the message interchange between objects (lifelines) [15], from EM Business Rule elements can be generated five UML Sequence model elements (Table 5): Execution Specification, Combined Fragment, Interaction Use, State Invariant and/or Destruction Occurrence.

Table 5. UML Sequence model elements [14, 15]

EM element	UML Sequence Model element	Description
Business Rules	Execution Specification	Represents a period in the participant's lifetime
	Combined Fragment	Defines a combination (expression) of interaction fragments. A combined fragment is defined by an interaction operator and corresponding interaction operands
	Interaction Use	Allows to use (or call) another interaction
	State Invariant	Represents a runtime constraint on the participants of the interaction
	Destruction Occurrence	Represents the destruction of the instance described by the lifeline

In case of UML Timing model, which shows interactions when a primary scope of the model is to reason about time [15], from EM Business Rule elements can be generated three UML Time model elements (Table 6): Duration Constraint, Time Constraint, Destruction Occurrence.

Table 6. UML Timing model elements [14, 15]

EM element	UML Timing Model element	Description
Business Rules	Duration constraint	Refers to a duration interval. The duration interval is duration used to determine whether the constraint is satisfied
	Time Constraint	Refers to a time interval. The time interval is time expression used to determine whether the constraint is satisfied
	Destruction Occurrence	Represents the destruction of the instance described by the lifeline

In case of UML Interaction Overview model, which identifies interactions through a variant of activity models in a way that sustains overview of the control flow [15], from EM Business Rule elements can be generated five Interaction Overview model elements (Table 7): Duration Constraint, Time Constraint, Interaction Use, Control Nodes.

Table 7. UML Interaction Overview model element [14, 15]

EM element	UML Interaction Overview model element	Description
Business Rules	Duration Constraint	Refers to a duration interval. The duration interval is duration used to determine whether the constraint is satisfied
	Time Constraint	Refers to a time interval. The time interval is time expression used to determine whether the constraint is satisfied
	Interaction Use	Allows to use (or call) another interaction
	Control Nodes	Used to coordinate the flows between other nodes. It includes: initial, flow final, activity final, decision, merge, fork, join

This wide group of elements can be generated because UML Interaction Overview model coordinates elements from activity and interaction models [15]:

- from the activity model: initial node, flow final node, activity final node, decision node, merge node, fork node, join node;
- from the interaction models: interaction, interaction use, duration constraint, time constraint.

All these UML models elements descriptions shows, how much knowledge about problem domain is stored in Enterprise model, how much information behind the EM Business Rule element is saved and also, how important that this stored information's quality and fullness should match all the different UML models approaches, so that suitable UML model could be generated.

4 Conclusions

The first part of the article handles with defining Enterprise model concept and with detailed explanation of Unified Modelling Language model generation from Enterprise model transformation algorithm, which is depicted by steps.

In the next part there are presented the Business Rule element's variations in different UML models, which were generated from Enterprise model, and how the quality of the business problem information, stored in knowledge-based Enterprise model makes the impact on these variations.

The described Business Rule element's example shows that if the high quality information stored in Enterprise model, then it is enough for UML models generating process. There is possible to confirm, that each element of each UML dynamic model can be generated from the Enterprise model using transformation algorithms only if high quality of the information in Enterprise model is ensured.

References

1. Butleris, R., Lopata, A., Ambraziunas, M., Veitaitė, I., Masteika S.: SysML and UML models usage in knowledge based MDA process. Elektronika ir elektrotechnika **21**(2), 50–57 (2015). Print ISSN: 1392-1215, Online ISSN: 2029-5731
2. Čyras, V.: Intelektualios sistemos. Vilnius: Vilniaus universitetas, Matematikos ir informatikos fakultetas, Programų sistemų katedra, 116 p. (2017)
3. Evensen, K.D., Weiss, K.A.: A comparison and evaluation of real-time software systems modeling languages. In: AIAA Infotech@Aerospace 2010, 20–22 April 2010, Atlanta, Georgia (2010)
4. IEEE Computer Society: Guide to the Software Engineering Body of Knowledge SWEBOK. Version 3.0. (2014). Paperback ISBN-13: 978-0-7695-5166-1
5. Gudas, S.: Enterprise knowledge modelling: domains and aspects. Technol. Econ. Dev. Econ. Baltic J. Sustain., 281–293 (2009)
6. Gudas, S.: Architecture of knowledge-based enterprise management systems: a control view. In: Proceedings of the 13th world Multiconference on Systemics, Cybernetics and Informatics (WMSCI 2009), vol. 3, pp. 161–266, Orlando, Florida, USA, 10–13 July 2009. ISBN -10: 1-9934272-61-2 (Volume III). ISBN -13: 978-1-9934272-61-9 (Volume III)
7. Gudas S.: Informacijos sistemų inžinerijos teorijos pagrindai. Vilniaus universiteto leidykla (2012). ISBN 978-609-459-075-7
8. Gudas, S., Lopata A.: Meta-model based development of use case model for business function. Inf. Technol. Control **36**(3) (2007). ISSN 1392 – 124X 2007
9. Küster, T., Lützenberger, M., Heßler, A., Hirsch, B.: Integrating process modelling into multi-agent system engineering. multi-agent and grid systems (2012). https://www.researchgate.net/publication/220535349_Integrating_process_modelling_into_multi-agent_system_engineering
10. Lopata, A., Veitaite, I., Gudas, S., Butleris, R.: CASE tool component – knowledge-based subsystem. UML diagrams generation process. Transformations Bus. Econ. **13**(2B) (32B), 676–696 (2014). ISSN: 1648 - 4460
11. Lopata, A., Ambraziūnas, M., Gudas, S., Butleris, R.: The main principles of knowledge-based information systems engineering. Electron. Electr. Eng. **11**(1)(25), 99–102 (2012). ISSN 2029-5731
12. Lopata, A., Ambraziūnas, M., Gudas, S.: Knowledge based MDA requirements specification and validation technique. Transformations Bus. Econ. **11**(1)(25), 248–261 (2012)
13. Lopata, A., Ambraziūnas, M., Gudas, S.: Knowledge-based MDA requirements specification and validation technique. Transformations Bus. Econ. **11**(1(25)), 248–260 (2012). ISSN 1648-4460
14. OMG UML: Unified Modelling Language version 2.5. Unified Modelling (2018). http://www.omg.org/spec/UML/2.5

15. UML diagrams (2018). https://www.uml-diagrams.org/
16. Veitaitė, I., Ambraziūnas, M., Lopata, A.: Enterprise model and ISO standards based information system's development process. In: Abramowicz, W., Kokkinaki, A. (eds.) BIS 2014. LNBIP, vol. 183, pp. 73–79. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11460-6_7

Visualization for Open Access – A Case Study of Karlstad University and the University of Makerere in Uganda

Prima Gustienė(✉)

Karlstad Business School, Karlstad University, Karlstad, Sweden
Prima.gustiene@kau.se

Abstract. Open Access (OA) research provides a platform to world-wide knowledge sharing, but the channels and possibilities are still limited. It is not always clearly defined how the process of publishing should take place. Information systems architectures are intrinsically complex engineering products that can be defined graphically on various levels of abstractions and represented using different aspects of the system. For that reason enterprise architecture is not easy to comprehend for different actors involved. Graphical representations of different business scenarios are critical to understand how different aspects of the system descriptions are analysed in relation to each other and how requirements from different perspectives are perceived as a whole. The goal of this paper is to introduce the modelling method for visualization of the process of publication of BSc thesis into institutional repositories. Universities need to investigate undergraduate students' publications of their graduate work (BSc thesis) to promote knowledge of university repositories. Two different modelling methods were used to visualize publishing process and two case studies at two different universities were made. The results indicated and motivated that SICM method had more semantic power to visualize business process in a more comprehensive way.

Keywords: Visualization · Open Access (OA) · Modelling methods
Enterprise system architecture

1 Introduction

Access to current, relevant and high quality research results has a big potential to drive research development. Open Access (OA) research provides a platform to world-wide knowledge sharing, but the channels and possibilities are still limited, especially in developing countries. It is not always clearly defined how the process of publishing should take place, especially, when different actors are involved. Universities need to investigate undergraduate students' publications of their graduate work (BSc thesis) to promote knowledge of university repositories for the benefit of later Master studies and also for the use by the community when working in various organizations after graduation. It is agreed today that graphical representations are more understandable then the requirements presented in natural language. Because of the ambiguity of natural language, the requirements presented in natural language, may lead to misunderstanding in

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 187–197, 2018.
https://doi.org/10.1007/978-3-319-99972-2_15

communicating different solutions. To make information exchange possible among different actors, it must be represented in communicable way [3]. Communication can be defined as a process of transferring and sharing knowledge in form of information flows from one actor to another as efficiently and effectively as possible. To understand the graphical representations of business processes and how these processes are related to other processes, all necessary architectural aspects should be integrated [8]. Different aspects should be synchronized to ensure consistency and completeness of the overall system specification [14]. Zachman framework [17] is one of the frameworks that comprises the enterprise architecture. It is a two-dimensional schema representing six perspectives. Integration of these perspectives contributes to understanding and enhancing business value to the enterprise [13]. Another view of enterprise architecture is presented by Lankhorst [10, 11]. It consists of organisational architecture representing organisational design, including organisational actors, their roles and relationships in organization. Information architecture represents data and its structure, process architecture represents business processes, application architecture represents the application logic and technical architecture represents technical infrastructure. Integration of heterogeneous architectural domains are critical for getting a holistic view of an enterprise. The quality of system specifications much depends on reaching semantic integrity of graphical representations. Semantic integrity is the degree to which graphical representations are perceived despite their complexity [8]. Graphical models have proven to facilitate understanding, communication and change management. It is critical to reach consistency between different aspects, as data and processes, in order to have a complete and integrated enterprise system specification.

Conceptual modelling can be seen as the key to a successful integration process, as it allows to focus on semantics and design of integration as opposed to implementation issues. It is the only way to visualize the system design and validate it against requirements. Conceptual modelling is a fundamental activity in requirements engineering [12]. It is an act of abstracting a model from a problem domain [11]. Visualization creates models with clarity as regards six questions or aspects of 'who', 'what', 'how', 'where', 'when' and 'why' [18]. The answers to these questions and integration of all the aspects provides us with a possibility to comprehend and understand more clearly the context as well as complexity of problem domain. According to Dietz [2], complexity can be mastered if two conditions are fulfilled. The first condition is to have a comprehensive theory about the things whose complexity one wants to handle. Another condition is that it is necessary to have a good methods and techniques based on that theory. To manage complexity it is critical to have an integrated method as well as a coherent, comprehensive, consistent and concise conceptual modelling approach [8]. To obtain value from graphical representations they must be integrated and semantically correct. With this we mean that it is important to choose semantically correct modelling method for visualisation of business process. The purpose of this paper is to show how to visualize the process of BSc thesis publishing on universities' repositories.

2 Research Method

A case study research method was applied incorporating two cases. Among the strengths of the case study methodologies is ability to study different aspects of a phenomenon, were the boundaries between the phenomenon (OA) and its context (practice and policies) is somewhere unclear [1, 15]. Interviews were used for the activities requiring establishment of practices and analysis of opinions, views and attitudes about OA at the participating institutions. Modelling approaches were used for visualisation and mapping the research information. As business processes - such as the publishing of students' final thesis into institutional repositories - consists of activities integrating both organizational and technical components, modelling outputs must be comprehensible not just for people with the IT system background, but also for other actors involved such as teachers, researchers, librarians, managers and administrators. Two different modelling methods were used for visualisation of BSc thesis publication process: 2conciliate method www.2c8.com and Semantically Integrated Conceptual Modelling Method (SICM) [4–7]. One of the reasons why the two modelling methods were used is that we also wanted to show advantages and disadvantages of these two modelling methods.

Karlstad University (KAU) Library and the University of Makerere, Uganda were selected as an appropriate cases to study the process of OA publications. KAU library was chosen, because KAU is one of the 47 universities that uses DiVA for research publications. At this university the analysis of the process of publication was made, the manual of the process was written in text and the process was also visualised graphically. The same modelling method was applied for the analysis and visualization of possible publication registration at Makerere University in Uganda. This university was chosen because of close research collaboration and because of the mutual involvement in the project "The role of Open Access in the maturing process of IT research education. Three case studies".

3 KAU OA Publication via Swedish National Portal DiVA

KAU has a vision to be an open university dedicated to the task of creating, integrating, applying, disseminating and preserving knowledge. The university supports free, new and grand thinking. The dialogue with community helps to develop knowledge that changes the world by challenging the borders of knowledge and structures. Karlstad University is in the process of designing its future support and policy for the processing of research data. The university promotes open access (OA) to scientific information to increase the use of research, access and transparency. "There are more political initiatives that support open access to research data. The aim is to enable the future re-use of data in research. In order to contribute to increased transparency, many funding bodies impose stricter conditions on researchers to report how the data they generate are processed, stored and made available to others," says Sofia Andersson, research advisor and project manager. During 2017, the Swedish Research Council was appointed the national coordinator of the implementation of open access to research data. The project also involves the National Library, the National Archives and higher education institutions. As a part of this project, a national network was created in partnership with the Swedish National Data Service for developing and implementing policies, routines and support at all Swedish higher education institutions. Karlstad University forms part of this national network for open research data.

3.1 DiVA-Academic Archive On-line

DiVA (Digital Scientific Archives) portal http://www.diva-portal.org is a finding tool for research publications and student thesis, written at 47 different universities and research institutions in Sweden. DiVA, an Academic Archive On-line, is a publishing system for research and students thesis. It is also a digital archive for long term preservation of publications. DiVA began its development in the year 2000 at Uppsala University Library in Sweden. All universities and publicly financed research institutions both in Sweden and abroad are welcome to join DiVA in its co-operative effort. The research publications and students thesis found in DiVA portal have been published and registered at the university or college of origin. There is bibliographic information in the database for every title and usually an abstract and a link to the full text at the university or college where it was published.

3.2 Visualisation of Publishing Process in DiVA at KAU with 2consiliate Method

It is critical to have a complete picture of a work-flow how the process of research publications take place: which actors are involved in every step, which information flow take place between actors. It is not enough to have just a manual with a text, but it is also important to have a graphical presentation. Natural language text can be understood ambiguously, and that may lead to misunderstanding but also having the graphical presentation of the process helps to comprehend it more clearly. As there are many actors involved in the publishing process, it is very important that the text is also visualized graphically. It is critical for all the actors involved in the process to know their roles and responsibilities. The actors should follow the process to fulfil the registration of the final thesis in DiVA correctly and in time.

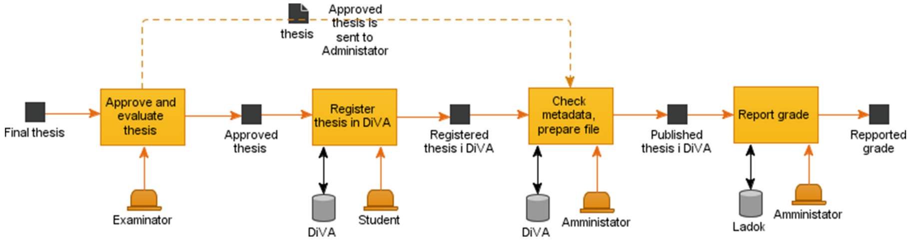


Fig. 1. The process of student thesis registration in DiVA. (Modified from Karlstad Universities guide for administration of student thesis https://www.kau.se/files/2016-10/Manual_uppsatsadministrat_DiVA2016.pdf) [9]

2conciliate is the method that support in the work of mapping, analysing and developing the organisation. The method is applied and integrated in the 2c8 Modeling Tool www.2c8.com. The 2conciliate method and the 2c8 Modeling Tool visualise organisations with models, symbols and relationships. The visualization facilitates communication, which leads that more people in the work group being able to participate. According to 2conciliate Business Solutions AB 2c8modelling tool is easy to understand, easy to learn and easy to use. Figure 1 presents the diagram, showing the process of student's final thesis registration in DiVA.

Figure 2 explains the symbols used in 2conciliate method modelling. There are six actors involved in the process: **Student, Supervisor, Examiner, Administrator, DiVA, LADOK** (LADOK is a database for reporting students' grades). The scenario of publication process is as following:

1. **Supervisor** gets the final thesis from the student, sets the grade and sends the final thesis to the **Examiner.** (The interaction between supervisor and examiner is not shown in the picture. The process begins when the Examiner gets the final thesis).
2. **Examiner** evaluates the thesis, sends the notification about the grade to the **Student** and to **Administrator** to approve the thesis registration in **DiVA.**
3. When the **Student** gets approved grade, he registers his thesis in **DiVA.** (Not obligatory action).
4. **DiVA** is sending e-mail to **Administrator** notifying the publication.
5. **Administrator** approves the publication of the thesis in **DiVA.**
6. **Administrator** reports the grade in **LADOK.**

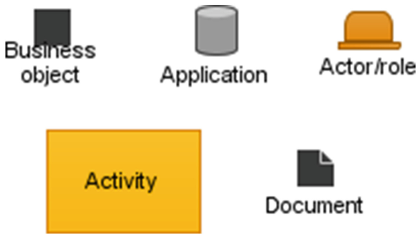


Fig. 2. Symbols used in 2conciliate method

When the modelling was done using 2conciliate method, the graphical presentation was analysed, in order to evaluate the semantic power of this modelling method. With semantic power we mean, the evaluation of the modelling method, how communicable and comprehensible the method and the model is. How much semantic power this modelling method has to represent all necessary aspects of business process. The biggest problem with 2conciliate method is that this method provides just sequential or in line modelling of actions. Objects (as information flows) is linked to the actions, as well as roles and actors. What was missing in this modelling method is possibility to visualise the interaction (loops) among different actors, showing their responsibilities, who is doing what and when. The method has no possibility to show data changes, when an action takes place. This is very important for value creation. For that reason another modelling method was used to model the same process. It was relevant to see if using another modelling method that is based on the other principles of modelling can present necessary aspects much better.

3.3 Visualisation of Publishing Process in DiVA at KAU with Semantically Integrated Modelling Method

Semantically Integrated Modelling method (SICM) [4–7] was used to model the same process. The essence of this method is that it is based on interaction principles among different actors involves, as well as the construct used, enables to integrate data and process using one diagram. This enables possibility to data changes and processes in one diagram [8]. Figure 3 presents KAU thesis approval and publication process using SICM modelling method.

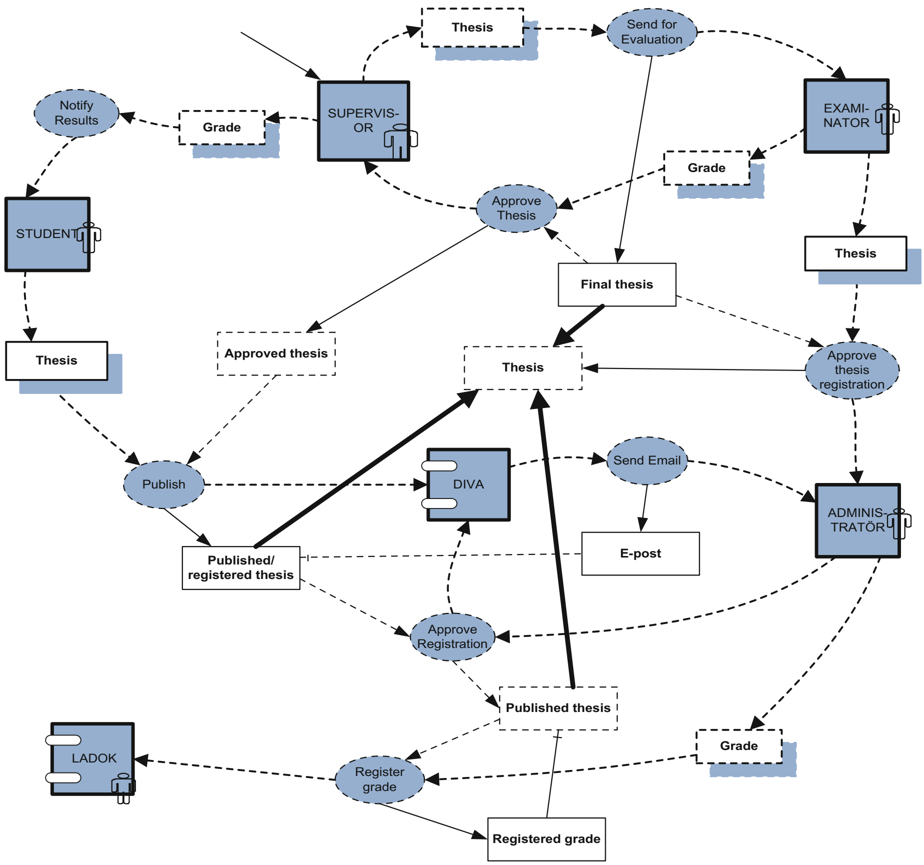


Fig. 3. KAU thesis approval and publication process

Figure 4 presents the main SICM notations used in modelling. There are following actors involved in the process: **Student, Supervisor**, **Examiner, Administrator, DiVA, LADOK** (**LADOK** is a database for reporting students' grades). The scenario of publication is as following:

1. **Supervisor** gets the final thesis from the **Student**, sets the grade and sends the final thesis to the **Examiner.**
2. **Examiner** evaluates the thesis, sends the notification about the grade to the **Supervisor,** who sends notification about grade to the **Student** as well to **Administrator** to approve the thesis registration in **DiVA.**
3. When the **Student** gets approved grade, he sends his thesis for publication to **DiVA.**
4. **DiVA** is sending e-mail to **Administrator** notifying the publication.
5. **Administrator** approves the registration of the thesis in **DiVA.**
6. **Administrator** registers grade in **LADOK.**

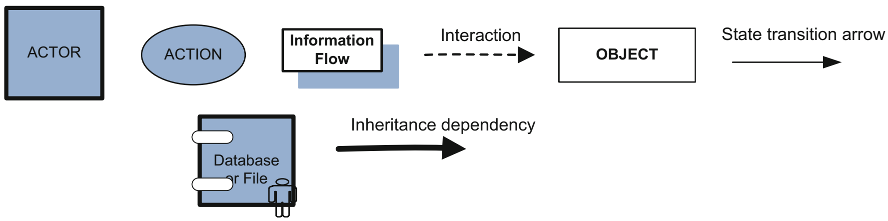


Fig. 4. Notation used for SICM method

NOTE: It is very important that the final thesis that the student publishes in DiVA is the same final thesis that the **Examiner** approved! The Administrator's action here is very important.

The essence of this modelling is based on interaction loops among different actors. Enterprise actors at conceptual level are represented by square boxes that can represent both organizational and technical components. Dynamic dependencies in SICM method are used to define relations between different actors, their actions and communication flows. Interaction is a dynamic relation. It is defined by communication action between two active actors. It indicates that one actor is dependent on other actor if some communication takes place. This is very important for the fulfilment of actors' responsibilities that are critical for value creation. At the same time, interaction is a strategic flow [16] dependency between actors. The communication dependency among enterprise actors are relevant for description of the 'who' perspective. Depending on whether there are flow elements in the action or not, the communication flow can be information, material or decision. Transition is a dynamic relation that in SICM indicates the fundamental changers of objects that take place during creation, termination or reclassification events. Transition dependencies represents the objective perspective of interaction. Actions indicate permissible ways for causing transitions of objects [8]. Inheritance dependency enables to follow different state changes of objects/data. When the action take place it should make changes to data otherwise man can doubt if the action is purposeful. One of the advantages of this method is that the method enables the integration of static (data) and dynamic aspects (interaction and transition) of the system. Another advantage is that everything is visualized using one diagram, one modelling construct.

3.4 Visualisation of Publishing Process in Dspace at Makerere University (MAK) in Uganda with Semantically Integrated Modelling Method

Makerere University (MAK) in Uganda was chosen for this case study because of the mutual project concerning open access. At Makerere University Library the registration of students' publications are not obligatory but they also have a database called Dspace, where it is possible to place research publications. The interview was done in Uganda MAK concerning the process of research publications. From the interview the procedure/scenario was recorded. According to this scenario the process was modelled and showed and discussed at Makerere University. The feedback was very positive, because the visualisation of the process was more comprehensible as the text.

The actors involved in the process are as following: **Student, Supervisor, Internal and External Examinators, Higher Degrees and Research Office (for examination), Directorate of Graduate Studies and Research. Directorate Graduate Research & Training (DRGT), Library.**

Scenario description is as following:

1. The **Student** submits the thesis to **Supervisor** to sign as a final thesis for examination.
2. The **Supervisor** signs the thesis and sends back to the **Student.**
3. The **Student** submits signed thesis to the **Higher Degrees and Research Office.**
4. **Higher Degrees and Research Office** organises a higher degrees and research committee meeting to assign internal and external examiners.
5. **Higher Degrees and Research** office submits the thesis to the examiners for examination.
6. Both **External and Internal Examiners** evaluates/puts the grade and sends the evaluation to the **Higher Degrees and Research Office.**
7. **Higher Degrees and Research Office** organises the defence of the thesis.
8. During the defence the student gets the marks from the panel in addition to the marks from the **Examiners.** (Steps 7 and 8 are not represented graphically in the picture. These steps belong to another process 'Defence of the Thesis).
9. After defence **Higher Degrees and Research Office** sends comments to the **Student** to finalise his thesis.
10. After the corrections, the student submits a hard copy and a soft copy of the final thesis to the **Directorate of Graduate Training and Research.**
11. **Directorate of Graduate Training & Research** submits the hard copy of the final thesis to the **Library**.

Figure 5 presents the visualisation of the publication process at MAK. In this scenario the process differs from the process of publication at Karlstad University. At MAK there are more actors involved, with more responsibilities. MAK university was pleased with the visualisation of publishing process, and they hope that such

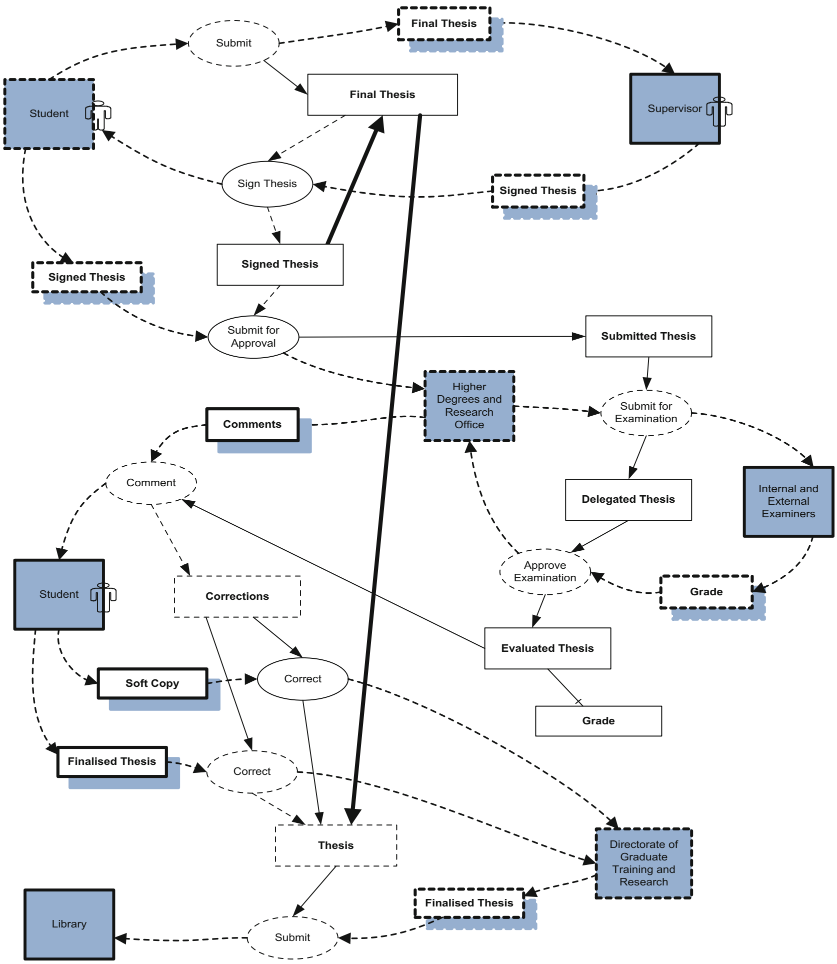


Fig. 5. MAK thesis approval process

visualizations of publications, will contribute to a better communication as well as it will contribute to extensions of Open Access policies at the MAK University.

4 Conclusion

Karlstad University promotes open access (OA) to scientific information to increase the use of research, access and transparency. Universities need to investigate undergraduate students' publications of their graduate work (BSc thesis) to promote knowledge of university repositories for the benefit of later Master studies and also for the use by the community when working in various organizations after graduation. The goal of this paper was to show how the visualization of business process can contribute to OA policy. It is very important that the process of publications is clearly defined and understood. It was shown how the publication of students' BSc thesis, were modelled, to increase communication and comprehensibility among different actors. As picture is worth thousand words, and as natural language is rather ambiguous, graphical presentations are necessary to complement natural language texts. Two different scenarios of publication were done, one at KAU and one at MAK. Two different modelling approaches were used for visualization of the same publishing scenario. We wanted to show which modelling method has more semantic power to visualise necessary aspects of the publishing process and why. Based on the results of the case studies done it appears that SICM modelling approach has more advantages in comparisons to 2conciliate modelling method. The biggest advantage of this modelling method is that it provides a new way of enterprise modelling and integration. It is a method for graphical analysis and design that enables reasoning about system architecture across organizational and technical system boundaries. The modelling method based on interaction loops among different actors enables the integration of data and processes. As 2conciliate modelling method uses in line or sequential way of modelling this modelling method has no sematic power to integrate data and processes, as well as to show data changes that are very important to comprehend business process as a whole. As a future work we plan to use modelling as a tool for the visualization of more publishing processes or other business processes that are important for OA policy.

References

1. Cavaye, A.L.M.: Case Study research: a multi-faced research approach for IS. Inf. Syst. J. **6**, 227–242 (1996)
2. Dietz, J.: Enterprise Ontology: Theory and Methodology. Springer, Berlin (2006). https://doi.org/10.1007/3-540-33149-2
3. van Griethuysen, J.J.: Concepts and Terminology for the Conceptual Schema and Information Base. Report ISO TC97/SC5/WG5. International Standard Organization, Genéve (1982)
4. Gustas, R.: Modeling approach for integration and evolution of information system conceptualization. Int. J. Inf. Syst. Model. Des. **2**(1), 45–73 (2011)
5. Gustas, R., Gustiené, P.: Conceptual modeling method for separation of concerns and integration of structure and behavior. Int. J. Inf. Syst. Model. Des. **3**(1), 48–77 (2012)
6. Gustas, R., Gustiené, P.: Semantically integrated conceptual modeling method and modeling patterns. In: Khosrow-Pour, M. (eds.) Systems and Software Development, Modeling, and Analysis: New Perspectives and Methodologies, pp. 1–33. IG Global, USA (2014)
7. Gustas, R., Gustiené, P.: Principles of semantically integrated conceptual modelling method. In: Shishkov, B. (ed.) BMSD 2016. LNBIP, vol. 275, pp. 1–26. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57222-2_1. ISBN 978-3-319-57221-5, ISBN 978-3-319-57222-2 (eBook)
8. Gustiené, P.: Development of a new service-oriented modelling method for information systems analysis and design. Doctoral thesis, Karlstad University Studies, Karlstad (2010)
9. Karlstad University: Karlstad Universities guide for administration of student thesis. https://www.kau.se/files/2016-10/Manual_uppsatsadministrat_DiVA2016.pdf

10. Lankhorst, M., van Leeuwen, D., Jonkers, H., ter Doest, H.: Enterprise architecture modelling: the issue of integration. Adv. Eng. Inf. **18**(4), 205–216 (2004). Special Issue on Enterprise Modelling and System Support
11. Lankhorst, M.: Enterprise Architecture at Work: Modelling, Communication and Analysis. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-642-29651-2
12. Nuseibeh, B., Easterbrook, S.: Requirements engineering: a roadmap. In: Proceedings of Conference on the Future of Software Engineering: International Conference on Software Engineering, pp. 35–46. ACM Press, New York (2000)
13. Rouse, W.B., Baba, M.L.: Enterprise transformation. Commun. ACM **49**(7), 66–72 (2006)
14. Whitten, J.L., Bentley, L.D.: System Analysis and Design Methods. McGraw Hill, London (1998)
15. Yin, R.: Case Study Research: Design and Methods. Sage, Thousand Oaks (2003)
16. Yu, E., Mylopoulos, J.: Why goal-oriented requirements engineering. In: Dobois, E., Opdahl, A.L., Pohl, K. (eds.) Proceedings of the Fourth International Workshop on Requirements Engineering: Foundations of Software Quality, Pisa, Italy, pp. 15–22. Manum University Press, Namur (1998)
17. Zachman, J.A.: Enterprise architecture: the issue of the century. Database Program. Des. Mag. **10**(3), 44–53 (1996)
18. Zachman, J.A.: A framework for information systems architecture. IBM Syst. J. **26**(3), 276–292 (1987)

Check for updates

A Model-Driven Approach for Access Control in Internet of Things (IoT) Applications – An Introduction to UMLOA

Mehreen Khan(✉), Muhammad Waseem Anwar, Farooque Azam, Fatima Samea, and Muhammad Fahad Shinwari

Department of Computer and Software Engineering, College of Electrical and Mechanical Engineering, National University of Sciences and Technology (NUST), Islamabad, Pakistan
{mehreen.khan16, fatima.samea16}@ce.ceme.edu.pk,
{waseemanwar, farooq}@ceme.nust.edu.pk,
m.fahad@nust.edu.pk

Abstract. The Internet of Things (IoT) is a collection of billions of devices attached to the internet that collect and exchange data using nodes, sensors, and controllers. The world is now continuously shifting from the traditional approaches to the IoT technology in order to meet the demands of modern technological advancements. However, the selection and implementation of right access control method in IoT applications is always challenging. In this context, OAuth is a renowned access control protocol in IoT applications. However, it is difficult to provide access control in IoT application through OAuth due to its implementation complexity. Therefore, there is a strong dire to introduce a model based approach that provide simple access control mechanism in IoT applications while preserving the major OAuth features. This article introduces **U**nified **M**odeling **L**anguage profile for **OA**uth (**UMLOA**) to model the access control requirements for IoT applications. Particularly, UMLOA is capable of modeling confidentiality, integrity, availability, scalability, and interoperability requirements in IoT applications. This provides the basis to transform the UMLOA source models into different target models (e.g. iFogSim etc.) for early verification of access control requirements. The applicability of UMLOA is validated through intelligent shipping container case study.

Keywords: UMLOA · MDA · IoT · Access control

1 Introduction

Internet of Things is a new computing paradigm and its use is increasing day by day in virtually all areas connecting a wide range of variety of devices such as appliances, automobiles, medical devices, sensors (RFID, NFC, etc.). However, there are several challenges and issues in IoT security. For example, William et al. [16] present a study by HP which reveals that 70% of the IoT devices do not encrypt their communications. Furthermore, 70% allow an unauthorized person to detect valid user accounts and 60%

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 198–209, 2018.
https://doi.org/10.1007/978-3-319-99972-2_16

of those with a user interface are vulnerable to different attacks such as denial-of-service, man-in-the-middle etc.

In this paper, we explore the problem of access control which is one of the most crucial aspect in IoT security. Access control is a way of limiting access to a resource and plays a critical role among different security mechanisms such as authentication, encryption etc. Implementation of access control in IoT applications is a complex process and a small error during implementation could make a critical resource accessible to standard users. The researchers in [16, 19] describes the managing mechanisms of access control. However, these are not applicable in IoT's domain due to some constraints such as Internet of things devices have limited network connectivity and have low computing capabilities etc.

Wu et al. [2] state that OAuth 2.0 protocol is a standard protocol and can be supported by number of programming languages and technologies. We adopt a model driven approach based on UML profile for access control with OAuth mechanism. Particularly, this article introduces **U**nified **M**odeling **L**anguage profile for **OA**uth (**UMLOA**) to model the access control requirements for IoT applications. UMLOA comprises several stereotypes to model complex access control requirements. This leads to transform the source models into target low level implementations like IFogSim. Furthermore, UMLOA can be used to automate the analysis of our approach by deploying it on real testbeds such as Fiware, FitIoT-LAB etc.

The paper is organized as follows: Sect. 2 present literature review and overview of the current research. In Sect. 3, proposed model driven approach is presented. In Sect. 4, validation is performed through intelligent shipping container case study. Finally, in Sect. 5, the discussion and benefits are presented.

2 Background and Related Work

The IoT is an emerging technology that is expanding quickly. Yu et al. [19], Gartner Inc. estimates that by 2020, number of deployed IoT devices will grow from 5 billion to 25 billion. Mineraud et al. [1] identified huge gaps in security with the potential to disrupt infrastructure and is plagued with malware and flaws e.g. self-storage, identification of management of IoT devices, manipulation of data in edge devices, secure authentication, support of constraint devices etc. Alqasem [12] explains that when dealing with security vulnerabilities and privacy, changing nature of IoT environment plays an important role because it consists of diverse technologies that leads to unknown risks and issues. Therefore, analyzing security requirements in complex IoT systems is essential.

Kolias et al. [3] discusses that the IoT botnets are recently emerging. One of them is Mirai botnet which attacked DNS and deduced the administrative credentials by means of brute force and caused major outages throughout the country. Therefore, IoT devices have minimal security and many flaws. Fernandes et al. [11] state that many systems are mass produced, therefore, they are similar in design because of the similar devices. One type of attack can be replicated across all. Upgrading or patching the operating system is not supported by vendors, because designing update system for IoT poses new challenges, they are once sold and done. These devices really don't have resources for traditional security methods as they are designed for low power and minimum memory. Therefore, feasible security measures for IoT implementation environment must be identified.

The proposed work by Peter et al. [14] includes various security features and each of these access control methods has its merits as well as demerits. For example use of hashing, encryption techniques, and threshold key cryptography provides counter measures against IoT attacks. One of them is SE-HAN, which uses a self-certified public key technique for key establishment and access control. But this technique doesn't provide much details regarding how it is efficient than others.

Ouaddah et al. [9] describes that OAuth is an access control framework that has recently gained a lot of attention. Related to large internet connected applications it was designed to address the issues of access control and privacy. This framework enables users to grant third party applications access to protected resources without revealing their credentials. Over one billion user accounts are OAuth based and are provided by major service providers such as Microsoft, Facebook and Google. Cirani et al. [8] discusses that in many research projects much effort is already in progress to implement OAuth over IoT protocols such as in EU project CALIPSO, BAC-net etc.

2.1 System Requirements for IoT in Access Control

Kim et al. [5], Minoli et al. [4] and Koivu et al. [15] states that IoT product development always require good practice and design with the CIA Triad. When developing products for the IoT, always consider in mind: confidentiality, integrity and availability. With confidentiality, the data should not be accessible to anyone without appropriate permissions. With integrity, the data should be stable and not mutable to anyone without appropriate permissions. With availability, the device should be available to anyone with appropriate permissions. We need a proactive approach to secure the IoT. Without this, attacks and malfunctions will outweigh any foreseeable benefits. Kim et al. [5] analyzed three main characteristics in internet of things security which includes heterogeneity, resource constraint and dynamic environment. Due to diverse hardware performances such as protocols, platforms and policies etc. common security service is the biggest problem of heterogeneity, which weakens interoperability. Due to lacking battery and capacity performance as stated by Jensen et al. [10], security services cannot be applied directly to IoT devices such as TLS, AED etc. therefore light weight algorithms need to be developed to increase memory, battery and CPU efficiency. Pasta et al. [7] publishes that IoT has a dynamic network topology due to bad connections and mobility. Large number of requests may be sent in very demanding cases such as smart city, smart home etc. due to numerous devices. Therefore, scalability and flexibility are required in IoT communication protocols. According to Kim et al. [5], IoT devices cannot save data due to low memory capacity. In case of missing the sensitive data, if cloud is out of order due to some reasons, the rescue service may be stopped that requires critical data. Therefore, device should have a backup and availability is necessary.

OAuth doesn't provide confidentiality and integrity. We must protect (HTTP) communications using additional layer, hence we have ensured both by adding transport layer security (TLS) end to end between actors. For solving the problem of heterogeneity, we ensured that by selecting OAuth, tokens provide all the information required for authentication. Therefore, it fits on low capable devices and results in strengthening interoperability. Availability is guaranteed by providing long expiration time of token and the device should be available to anyone with appropriate permissions. This protocol provides flexibility and scalability for clients by detaching authorization entities from the rest of the entities, therefore, no need to operate directly on device.

The proposed UMLOA (Sect. 3) is based on Sciancalepore et al. [6], a well-known reference architecture as shown in Fig. 1. Particularly, recently emerging access control attributes are considered in the development of UMLOA. Our proposed model driven UMLOA approach offers the access control mechanism for internet of things. We will show how the reference architecture concepts and properties can be used to simplify analysis and modeling.

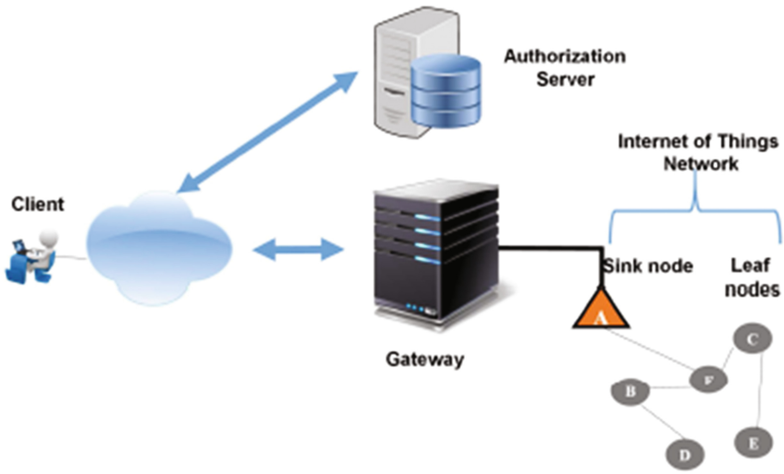


Fig. 1. Reference architecture using OAuth in Internet of Things network [6]

2.2 OAuth 2.0

OAuth 2.0 is new framework evolution as presented by Emerson et al. [18]. It is a token based authorization and authentication. OAuth architecture consists of four actors as follows: (1) **Resource Owner** – RO can grant access to a protected resource and it's typically an end user. (2) **Authorization Server** – AS authenticates a RO and issues access tokens after getting proper authorization. (3) **Client** – an application requesting access to protected resources. (4) **Resource Server** – RS is hosting the protected resources.

Gabriel et al. [17] states that with in IoT environments to solve issues of access control, OAuth comes as a good candidate and is used by large enterprises such as Google, Facebook, Windows Live etc. as already mentioned above. OAuth provides access to protected resources in the following way.

1. First Client contacts the Resource Owner.
2. The Resource Owner Grants the authorization code.
3. The Client delivers the authorization code to Authorization Server.
4. After verifying the authorization code, the Authorization Server releases an access token to the Client.
5. The Client passes the access token to the Resource Server.
6. The Resource Server validates the access token and provides the protected resource.

3 Proposed Profile

Model Driven Architecture (MDA) is well-known application development approach that significantly simplify the design and verification of system under development [20]. Consequently, it is commonly used in the development of complex applications. In MDA, UML Profile is a kind of Unified Modeling Language extension mechanism. Some of the language elements are specialized such as icons and symbols. UML profile comprises following major elements: (1) Meta-class shows the classes in UML profile. (2) Interface point out the all interfaces that are included in a profile. (3) An operation is the meta-class that shows the functionality of profile. (4) Stereotype tagged values are the basic mechanisms of UML profile. UML consists of dynamic modeling and static modeling. Dynamic modeling shows the behavioral aspects of a system and can be described through state machine diagram and sequence diagrams. Structural modeling consists of structural aspects of a system and can be described through class diagram defined in terms of attributes and relationships. In this paper, we take class diagram, stereotypes and tagged values for a structural view. The proposed UMLOA is shown in Fig. 2.

3.1 UMLOA Description

In this section, we have presented the UMLOA stereotypes. The description, a base class and tagged values of each stereotype is described. There are 15 stereotypes for managing access control for protecting resources as follows:

Client Stereotype.
Description: It is third-party application, which on behalf of Resource Owner is requesting resources which belong to internet of things network
Base Class: Class
Tagged Values: It has two tagged values which are as follows

(1) Client ID: EInt[1] is public identifier for apps. (2) *Client Secret: EString[1]* is known only to AS and app.

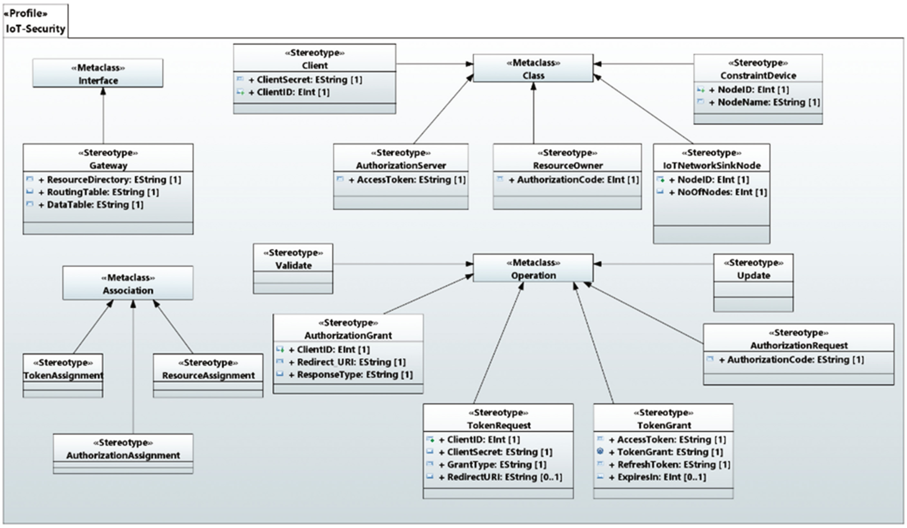


Fig. 2. Proposed Unified Modeling Language profile for Access Control using OAuth (UMLOA)

Resource Owner Stereotype.
Description: It can grant access to a protected resource and it's typically an end user.
Base Class: Class
Tagged Values: Resource Owner contains tagged value called *Authorization Code: EInt[1]* which has a type EInteger.

Authorization Server Stereotype.
Description: It authenticates a RO and after getting proper authorization it issues access token
Base Class: Class
Tagged Values: it contains tagged value called *Access Token: EString[1]*. It is a JWT or a opaque string used by an application to access API.
Gateway Stereotype.
Description: It implements an OAuth 0.2 Resource Server and is a primary element of architecture which offers security functionalities such as access control, authentication, resource tracking etc.
Base Class: Interface
Tagged Values: Gateway Stereotype have three tagged values of type EString, which are as follows

(1) Resource Directory: EString[1] Resource path is added here. (2) *Routing Table: EString[1]* represents active nodes. (3) *Data Table: EString[1]* represents record in a table with different identification of data also works as a cache.

IoT Network Sink Node Stereotype.
Description: Many constraint devices are integrated in internet of things network. These devices able to sense the surrounding environment, gather data and send them to a network co coordinator called a sink node and this sink node is attached to the Gateway which acts as a resource server.
Base Class: Class
Tagged Values: IoT Network Sink Node Stereotype have two tagged values of type EInt and EString, which are as follows

(1) NodeID: EInt[1] is a node identifier in IoT Network. (2) *NoOfNodes: EString[1]* mentions the number of Nodes in IoT Network.

Constrained Devices Stereotype.
Description: IoT Devices with limited memory and power resources.
Base Class: Class
Tagged Values: Constrained Device Stereotype has tagged value of type EInt called Node Identifier represented as *NodeID: EInt[1]*
Resource Assignment Stereotype.
Description: It is used to assign a resource
Base Class: Association
Token Assignment Stereotype.
Description: It is used to assign a token
Base Class: Association
Token Request Stereotype.
Description: It is used to access a token
Base Class: Operation
Tagged Values: Token Request Stereotype contains four tagged values which are as follows:

(1) *ClientID: EInt[1]* is public identifier for apps. (2) *ClientSecret: EString[1]* is known only to AS and app. (3) *GrantType: EString[1]* are of different types for acquiring an access token for different usecases such as implicit grant, client creditionals grant etc. (4) *RedirectURI: EString [1]* is the callback entry point of application that prevents tokens from being intercepted.

Token Grant Stereotype.
Description: Represents response from a token, *Refresh token* is used to get a new access token in case of token expiration.
Base Class: Operation
Tagged Values: Token Request Stereotype contains four tagged values which are as follows:

(1) *AccessToken: EString[1]* a JWT signed with AS private key. (2) *TokenType: Estring[1]* with the value Bearer (3) *RefreshToken: EString[1]* is an encrypted payload (4) *ExpiresIN: EString[0..1]* representing TTL of access token.

Authorization Request Stereotype.
Description: It is used to request authorization from RO
Base Class: Operation
Tagged Values: Authorization Request Stereotype contains three tagged values client identifier of type EInteger, ResponseType with code of type EString and RedirectURI with the user to redirect back to of type EString, which are as follows:

(1) *ResponseType: EString[1]* (2) *ClientID: EInt[1]* (3) *RedirectURI: EString[0..1].*

Authorization Grant Stereotype.
Description: Represents response from RO
Base Class: Operation
Tagged Values: Authorization Grant Stereotype contains two tagged values which are as follows:

(1) *Code: EString[1]* authorization code (2) *State:EString[1]* optional but recommended.

Authorization Assignment Stereotype.
Description: It is used to assign an authorization
Base Class: Association
Update Stereotype.
Description: It is used to update the Data table in Gateway
Base Class: Operation
Validate Stereotype.
Description: It is used to validate the token in Gateway
Base Class: Operation

4 Case Study

We first introduce a design problem with its access control requirements mentioned above in (Sect. 2.1) and domain problem specifications that serve as a running example. Here, we have considered a subset of requirements of a system for IoT-Intelligent Shipping Containers. This use case involves containers and tracking environment factors such as humidity and temperature to ensure the delivery of contents to their destination in good condition. The containers are equipped with number of sensors e.g. location, temperature humidity, shock connected to embedded microprocessors. Containers also communicate with cloud services for reporting and analytics purposes. Ships and containers maintain constant network connectivity with cloud services while they are docked at the port. For monitoring the cargo status the ship needs to access the sensors en route, even in a location where network connectivity is not available. En route ship needs to update sensors continuously and in case where cargo is at some risk, the ship may alter plans of delivering the cargo and for making offline decisions when a network is not available, containers need to be able to cache trust validation. The model of intelligent shipping container through UMLOA is show in Fig. 3.

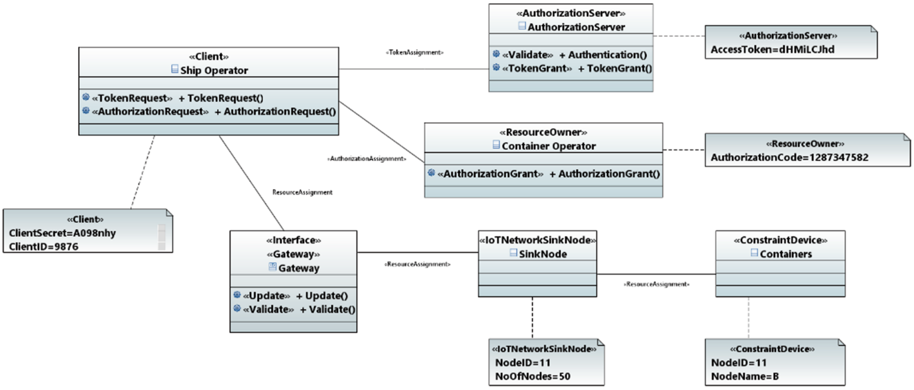


Fig. 3. Model of intelligent shipping container through UMLOA

According to requirements, the Ships act as a Client, Containers and their individual sensors as Resource Servers. The Container Operator is the Resource Owner and Ship Operator is the Requesting Party. These actors establish an end to end transport layer security (TLS). The Exposed Resources are registered with the Authorization Server on behalf of the container operator. Connectivity is available when the shipped is docked, in order the validate access request, the container will be able to interact with the authorization server. Even when the connectivity is not available containers are expected to validate the request, we can achieve this by either establishing a long enough expiration time of a token so that containers can provide service during the entire trip. Sensor information for the containers can be accessed by the ship and when connectivity is available, it might decide to refresh them.

The Gateway shows the most significant point of contact between the echo systems of IoT and the OAuth 2.0. More precisely, it also takes charge when it comes to storing the information about resources that are available thus providing system functionalities such as data caching and security functionalities such as establishment of transport layer security(TLS) with a client, access control and through resource discovery procedure tracking of available resources. Furthermore, these resources are also shown to the remaining architecture using the authorization primitives of OAuth 2.0. To attain what it was designed to do, three different data structures are used namely the Data Tables, Routing Tables and Resource Directory. The constant interaction between the resource servers and the IoT's sink node (IoT network coordinator) make sure that the above-mentioned data structures always have data. Identifiers of all the active nodes (sensors) in the IoT network are placed in the routing table. These nodes are also stored with the information of path that could be used for the communication between each of them. The gateway starts a procedure for the discovery of the resources, whenever new data is added to the routing table. The resource path obtained is then added to the Resource Directory. Whenever any kind of new data is obtained from the IoT network, the Data Table is updated by the gateway. In particular, every record in the table represents a different identification of a data for a resource. Data Table also works as a cache whose purpose can also be to answer all the external requests. This then leads to the limited consumption of energy within an IoT network. This shows the clear interaction between an IoT network and a gateway.

On the left side of Fig. 3, a client (ship operator) is shown that required the access to the available network. To make this happen the network starts the authorization process which is based on the workflow of the OAuth 2.0. This all starts off with a request being sent to the authorization server. This server then requests for the authorization from the container. The RO (Container Operator) then types in the required credentials to authorize the access and grants an authorization code. This then leads to the authorization server generating an access token and sending it to the client. The authorization server provides the access token attribute. Natively our proposed approach supports any token format such as JWT's, PoP and bearer tokens. If bearer tokens are used then at reception time there is need to process this token immediately by gateway as there is no cryptography technique is used to protect the content of token. With JWT's, for validation the content of token it contains a sign field. Asymmetric and symmetric cryptographic techniques can be used for this purpose. The case considered here is Asymmetric cryptography. After the tokens are triggered and sent, these tokens processed and validated by the gateway and come up with the answer accordingly. Hence, the requested resource is delivered to the client securely.

5 Discussion

According to Chernyshev et al. [13], IoT systems are never predictable and simple due to number of entities and the complexity of communication lines. It's highly impractical to create a realistic environment for testing, therefore, simulation can be carried out for early verification. OAuth uses asymmetric cryptography scheme which enables fine-grained access control. In this regard, UMLOA provides high abstraction layer for OAuth. Consequently, the complexity of OAuth protocol is significantly reduced through UMLOA. Furthermore, it enables the automatic generation of target OAuth implementations from the source UMLOA models. Hence, through this, we can increase quality, remove the friction and make a better decision by rapidly prototyping. Furthermore, we can be able to give practical feedback to users when designing the system. This allows the designers to determine correctness and efficiency before the system is actually constructed. UMLOA also improves the development time and reduced cost. This work is still under progress to validate the well-formedness and evaluate performance of the proposed UMLOA framework. We are working on transformation engine to automatically generate different target models (e.g. iFogSim etc.) from UMLOA source models for early verification.

6 Conclusions

This paper presents **U**nified **M**odeling **L**anguage profile for **OA**uth (**UMLOA**) to enable access control in IoT (Internet of Things) context by using the features of OAuth 2.0 protocol. UMLOA is easily deployable and manages access control mechanisms in a concrete internet of things scenarios and satisfies the special requirements of IoT devices in terms of confidentiality, integrity, availability, scalability and interoperability. The proposed approach comprises fifteen stereotypes to represent access control requirements in IoT applications through models. This provides the basis to transform the UMLOA source models into different target models (e.g. iFogSim etc.) for early verification of access control requirements. Consequently, UMLOA significantly reduces the implementation complexity of access control mechanism in IoT. The applicability of UMLOA is validated through intelligent shipping container case study.

This work is still under progress and we are working on the transformation engine to automatically generate several target models (e.g. iFogSim etc.) from UMLOA source models for early verification.

References

1. Mineraud, J., Mazhelis, O., Su, X., Tarkuma, S.: A gap analysis of Internet-of-Things platforms. J. Comput. Commun. **89**(C), 5–16 (2016)
2. Wu, X., Steinfeld, R., Liu, J., Rudolph, C.: An implementation of access-control protocol for IoT home scenario. In: IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS) (2017)
3. Kolias, C., Kambourakis, G., Stavrou, A., Voas, J.: DDoS in the IoT: Mirai and other botnets. Computer **50**(7), 80–84 (2017)
4. Minoli, D., Soharby, K., Kouns, J.: IoT Security (IoTSec) considerations, requirements, and architectures. In: IEEE Annual Consumer Communications and Networking Conference (CCNC) (2017)
5. Kim, Y.-G., Oh, S.-R.: Security requirements analysis for the IoT. In: International Conference on Platform Technology and Service (PlatCon) (2017)
6. Sciancalepore, S., Piro, G., Caldarola, D., Boggia, G., Bianchi, G.: OAuth-IoT: an access control framework for the IoT based on open standards. In: IEEE Symposium on Computers and Communications (ISCC) (2017)
7. Pasta, B., Pieres, P.P.: Design and analysis of IoT applications: a model-driven approach. In: 4th International Conference on Dependable, Autonomic and Secure Computing (2016)
8. Cirani, S., Picone, M., Gonizzi, P., Veltri, L., Ferrari, G.: IoT-OAS: an OAuth-based authorization service architecture for secure services in IoT scenarios. IEEE Sens. J. **15**(2), 1224–1234 (2015)
9. Ouaddah, A., Mousannif, H., Elkalam, A.A.: Access control in the Internet of Things: big challenges and new opportunities. J. Comput. Netw. **122**(C), 237–262 (2017)
10. Jensen, J., Jaatun, M.G.: Security in model driven development: a survey. In: IEEE International Conference on Availability, Reliability and Stability (2014)
11. Fernandes, E., Rahmati, A., Eykholt, K., Prakash, A.: Internet of Things security research: a rehash of old ideas or new intellectual challenges? IEEE Priv. Secur. **15**(4), 79–84 (2017)
12. Alqasem, I.: Privacy and security requirements for Internet of things (IoT). In: ICSE Companion (2014)
13. Chernyshev, M., Baig, Z., Bello, O., Zeadally, S.: Internet of Things (IoT): research, simulators, and testbeds. IEEE Internet of Things J. (2017)
14. Peter, S., Gopal, R.K.: Multi-level authentication system for smart home-security analysis and implementation. In: International Conference on Inventive Computation and Technology (2016)

15. Koivu, A., et al.: Software security considerations for IoT. In: IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) (2016)
16. Stout, W.Ms., Urias, V.E.: Challenges to securing Internet of Things. In: IEEE International Carnahan Conference on Security Technology (ICCST) (2016)
17. Gabriel, H., Timoteo, R.: Security analysis of a proposed Internet of Things middleware. J. Cluster Comput. **20**(1), 651–660 (2017)
18. Emerson, S., Choi, Y.K., Hwang, D.Y., Kim, K.S., Kim, K.H.: An OAuth based authentication mechanism for IoT networks. In: IEEE International Conference on Information and Communication Technology Convergence (ICTC) (2015)
19. Yu, T., Sekar, V., Seshan, S., Agarwal, Y., Xu, C.: Handling a trillion (unfixable) flaws on a billion devices: rethinking network security for the Internet-of-Things. In: Proceeding 14th ACM Workshop on Hot Topics in Networks, p. 5 (2015)
20. Anwar, M.W., Rashid, M., Azam, F., Kashif, M.: Model-based design verification for embedded systems through SVOCL: an OCL extension for SystemVerilog. J. Des. Autom. Embed. Syst. **21**(1), 1–36 (2017). https://doi.org/10.1007/s10617-017-9182-z

Check for updates

A Workflow-Based Large-Scale Patent Mining and Analytics Framework

Mustafa Sofean(✉), Hidir Aras(✉), and Ahmad Alrifai(✉)

FIZ Karlsruhe,
Hermann-von-Helmholtz-Platz 1, 76344 Eggenstein-Leopoldshafen, Germany
{mustafa.sofean,hidir.aras,ahmad.alrifai}@fiz-karlsruhe.de
http://www.fiz-karlsruhe.de

Abstract. The analysis of large volumes and complex scientific information such as patents requires new methods and a flexible, highly interactive and easy-to-use platform in order to enable a variety of applications ranging from information search, semantic analysis to specific text- and data mining tasks for information professionals in industry and research. In this paper, we present a scalable patent analytics framework built on top of a big-data architecture and a scientific workflow system. The framework allows to seamlessly integrate essential services for patent analysis employing natural language processing as well as machine learning algorithms for deeply structuring and semantically annotating patent texts for realizing complex scientific workflows. In two case studies we will show how the framework can be utilized for querying, annotating and analyzing large amounts of patent data.

Keywords: Patent analysis · Text- and data mining
Big data analytics · Visual workflow systems

1 Introduction

A patent is an exclusive right for an invention being either a product, a process offering a new way of doing something or a new technical solution to a problem. Patent documents are important intellectual resources that can help to protect the interests of individuals, organizations and companies. A patent document contains much more detailed information about a technology than any other type of scientific or technical publication. It is also a unique source of information, as it is estimated that on average about 70% of the information enclosed in patents is never published anywhere else. Hence, patent analysis plays an increasingly important role in defining business strategies and supporting decision-making in companies. For example, organizations are interested in analyzing patents to determine the novelty of patents, identify patent trends, predict technological developments in a particular area, find infringements, technological vacuums and hotspots, or important market competitors, etc. Various methods involving machine learning and text mining have been proposed in order to extract value from patent information [7]. The most recent change in the field of machine

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 210–223, 2018.
https://doi.org/10.1007/978-3-319-99972-2_17

learning has been the widespread use of neural networks e.g. Deep Learning (DL) for resolving complex tasks. Word embeddings are a well known method based on DL, where words are represented as a low dimensional real-valued vector enabling convenient and real world usage of neural networks [9] for text analysis and natural language processing (NLP).

In this paper, we describe a scalable patent mining and analysis framework for the efficient, flexible and large-scale analysis of patent data by using enhanced text and data mining methods. Our framework allows for the deeply structuring, analyzing and semantic enrichment of single patents as well as a large collection of patent documents via dedicated patent analysis services such as for claim structure recognition, detailed description segmentation, key term extraction, and named entities detection, e.g. chemical and drug entities. Moreover, our framework allows to develop efficient Spark-based services and data mining pipelines for scalable patent analysis employing natural language processing and machine learning methods such as topic modeling and extraction. Besides that, our aim is to enable research and development of new methods for data analysis focusing on scientific and technical information by supporting a variety of user and customer use cases in research and industry via scientific workflows. Hence, our framework combines and seamlessly integrates visual workflow development capabilities with a big data processing architecture based on Apache Hadoop and Apache Spark.

The remainder of the paper is organized as follows. Section 2 presents the related work. In Sect. 3, we describe our framework concept and the essential architectural parts of our system. Section 4 presents two case studies for big-data analytics of large-scale patent data utilizing our framework. Finally, Sect. 5 provides concluding remarks and directions for future studies.

2 Related Work

Existing methods for patent analytics focused mainly on technology trends analysis [2], technology forecasting [4], or the detection of new opportunities for technologies [6], etc. The work in [7] provides an overview of previous research for text mining, patent analysis based on visual interaction, and a taxonomy that classifies these approaches. Additionally, various clustering algorithms such as based on reinforcement learning [10], support vector clustering [11], and matrix factorization [12] were applied to patent data.

The work described in [8] examined the effectiveness of deep learning for the task of keyword extraction from patent texts based on the Skip-gram [9] model. In [13], the authors proposed a big data framework for analyzing patents for supporting strategic R&D decision-making processes. While the latter focuses on the architecture of the big data framework in order to enable the utilization of various patent analysis methods, our focus herein is on large-scale text and data mining methods for patent analytics. In contrast to previous approaches for patent analysis our aim was to

1. integrate workflow processing, scalable analysis and large-scale text-and data mining tasks seamlessly,

2. support single and multi-document analysis based on deeply structured patent text documents,
3. enable semantic enrichment, analysis and a highly interactive UI in one platform.

3 Framework Overview

In the following, we will describe the main parts of our large-scale patent analytics architecture. The framework allows to execute distributed and parallel services for analyzing big data of patents employing a highly interactive visual interface based on scientific workflows. Figure 1 shows the main parts comprising the workflow processing and the user interface, the search and retrieval infrastructure, and the big data processing and analysis services utilizing Hadoop and Spark jobs in a cluster.

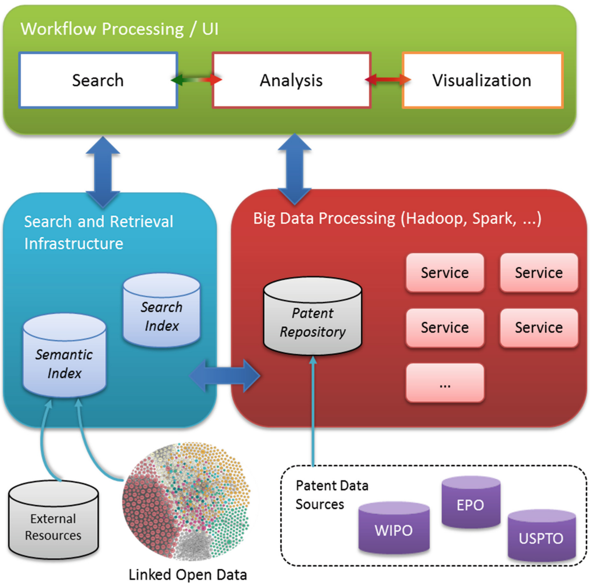


Fig. 1. The platform system architecture.

3.1 Data Sources and Patent Repository

FIZ-Karlsruhe provides about 200 different databases with patents and added value accessible through the STN Search Engine[1]. STN is an information service for scientific and technical information focusing on patents and offering online access to high-quality patent data on a neutral platform. It also contains

[1] http://www.stn-international.de.

full-text databases with patent specifications published by the most important patent authorities and offices. In the above described system, scalable distributed services based on Hadoop are employed in order to extract, transform and load the patent data from a variety of data sources into our distributed data repository. In an internal ETL process the patent data is gathered from the individual sources and transformed to a standard target format. We use Apache Phoenix[2] as a suitable NoSQL database for storing, querying and processing patent documents and additional meta-data for analytics purposes. The patent repository is also used for creating the patent search index for the unstructured fulltext of the patent documents as well as the structured part comprising important meta-data fields such as bibliographic information, legal status, classification codes, etc.

3.2 Scalable Analytics Services with Map-Reduce and Spark

Large-scale patent analysis has much higher requirements than single document, as it is needed to apply scalable and distributed analytics services for extracting hidden knowledge from a large collection of patent documents. Depending on the complexity and the duration of the patent analysis task we employ Hadoop map-reduce jobs or Spark-based analytics services for calculating the required models. In our scenario, we execute map-reduce jobs for performing basic tasks such as ETL, indexing or semantic annotation, while complex tasks which make use of advanced machine-learning and data mining algorithms benefit from the in-memory processing and efficient data handling and analytics capabilities of Spark[3].

Basic Analytics Tasks. A crucial task before advanced analysis processes on texts can be performed is to deeply structure and enrich the unstructured part of the patent data. Therefore, we first semantically segment the unstructured textual parts of a patent document. Afterwards, a sequence of semantic enrichment methods are applied for adding added value such as keyterms, chemical entities, etc. to the patent documents. The implemented services are described more detailed in Sect. 4.1.

Enhanced Big Data Analytics. We designed and developed efficient large-scale patent text and data mining (TDM) services by employing the Apache Spark framework. Spark was developed as an alternative execution engine for distributed processing and analysis, capable of performing faster computation by using in-memory primitives. The developed Spark-based services have been integrated into the framework and can be executed by clients in their end user tasks and workflows. We used the Latent Dirichlet Allocation (LDA) machine learning algorithm in our analytics services for extracting the list of most relevant topics from the collection of patent texts, and k-Means for partitioning the patent

[2] https://phoenix.apache.org.
[3] https://spark.apache.org/.

texts into clusters depending on their similarity. For clustering the metadata of the patents such as patent assignees, citations, and the IPC patent classification codes we employ k-Means. In addition, a word count service is used to get the top terms and phrases from all or specific segment/s of a patent text that is produced by our map-reduce-based service. The word count service can be also executed on the meta-data in order to get the most active inventors, companies, IPCs, or countries for the purpose of statistic analysis.

3.3 Workflow-Based User Interface

Scientific workflow systems allow to encode and document individual scientific processes and approaches in an understandable and transparent manner. For our patent analysis scenarios, workflows can be used to efficiently realize different data analytics use cases comprising a search phase, one or more analysis steps and additional post-processing and results interaction. In our system we used the Konstanz Information Miner framework (KNIME)[4] as a front-end tool in order to create distinct patent analytics workflows. A workflow in KNIME consists of several nodes belonging to various categories (readers, manipulators, learners, predictors, writers, etc.) that are connected via ports. A connection can either transfer data or generate models that describe extracted information from the input data such as learned predictors or models. In particular, we allow for the creation of workflows in KNIME for searching in a patent repository, executing a specific big-data analytics service, and visualizing and interacting with the results efficiently. The generated results can be explored and analyzed visually by means of automatically created tables, charts and diagrams or reports that describe the insights gathered from the examined result set of documents or an entire document collection.

As introduced above, scalable services are employed in order to perform fast an efficient pre-processing, basic (statistical) analysis or apply enhanced machine-learning or text mining algorithms to collections of patent data. All services that are developed in the Java language can be deployed as Jar files and stored in HDFS. We created a designated KNIME node called Generic Service Executor that uses the YARN REST API to submit, monitor, and kill any service such as ETL, annotation, or analytics service. In order to run any service, the end user only needs to select the name of the service and the related parameters as well as set the Hadoop configuration parameters. Depending on the complexity and type, the services are either realized via the Map-reduce or the Spark framework. Our developed Spark services are available on the Spark cluster as separated Jar files. The KNIME platform has a Spark Executor extension comprising a set of nodes used to create and execute previously defined Apache Spark applications by using the spark-jobserver[5], providing a RESTful interface for submitting and managing Apache Spark jobs. In order to run our own Spark

[4] https://www.knime.com/.
[5] https://github.com/ooyala/spark-jobserver.

services from a KNIME workflow, we utilized the Spark-java-Snippet[6] nodes to call the services, as they allow executing arbitrary Java code to manipulate or create Spark RDDs (Resilient Distributed Dataset). All our services provide results as JSON, which can be used efficiently for post-processing as well as visualization purposes.

4 Large-Scale Patent Data Analysis

We adopted various patent processing and analysis services for gathering, analyzing, discovering new technologies and trends in a collection of patent documents, as well as visualizing the results in different forms. A patent document often contains dozens of fields that can be grouped into two categories: First; structured fields (meta-data), which are uniform in semantics and format such as patent number, inventor/s, citation/s, filing date, issued date and assignee/s. Second; textual fields, that consist of title, abstract, claims, and the detailed description of the invention. The latter includes the technology area, background, summary, embodiments, and the description of figures and drawings of the invention.

4.1 Semantic Enrichment

The first scalable patent service we have developed can be used to structure the description part of patent text into semantic segments such as the technical field, background, summary, description of figures, and the embodiments. We developed a MapReduce-based annotation tool called PatSeg which is based on a combination of text mining techniques: machine learning (SVM), a rule-based algorithm, and heuristics which are utilized to identify the semantic segments in the description text of a patent document. The performance of our methods achieved up to 94% of accuracy [16]. Another scalable MapReduce-based service called Claim Structure Recognition (CSR) is able to automatically identify the complete claim hierarchy within patent claim texts [17]. The service is based on the Information Processing Framework (UIMA)[7] achieved up to 93% of accuracy in an expert-based evaluation.

Besides annotation services for deeply structuring patent text, several enrichments based on text mining methods and natural language processing have been realized. Keywords extracted from a patent document are of great benefit for search and content analysis. We developed a scalable keywords extraction service for extracting the significant words and phrases from a patent text. The service is based on a hybrid data pipeline taking into account both linguistic, statistical properties, and applying additional heuristic rules, e.g. based on position, length or layout [18] for identifying invention-relevant words and phrases.

Patents from the chemical or pharma domain are the main source of information for chemical analysis and drug discovery [19]. Therefore, we developed a

[6] https://www.knime.com/nodeguide/big-data/spark-executor/modularized-spark-scripting.

[7] https://uima.apache.org/.

service to enrich chemical patents with chemical entities employing suitable tools e.g. chemical tagger. Herewith it is possible to link external knowledge bases such as ChEBI to our data and allow for querying enriched information structures for answering complex scientific questions for the chemistry or bio-pharma domain [20]. We used the Open-Source Chemistry Analysis Routines (OSCAR4)[8] toolkit for the recognition of chemical named entities in patent texts. We then mapped each entity to external resources by utilizing the UniChem API[9].

4.2 Multi-patent Documents Analysis

The World Intellectual Property Organization (WIPO) has developed guidelines for patent search and analysis in order to create Patent Landscape Reports (PLRs) that supports informed decision-making. The guidelines are designed to efficiently address the concerns associated with making high stakes decisions in various areas of technology, increasing the related degree of confidence [21]. In accordance with these guidelines, we extracted the accurate technical information that represents the subject matter of an invention from important sections of the patent text employing methods for tracking the trend of a specific area. We identify relevant topics and uncover related research frontiers by focusing and investigating in a specific technological area.

Pre-processing Patent Text. In addition to understanding the structural complexity of patent texts by using PatSeg and CSR, we need to reduce the lexical complexity of patent texts. Therefore, Natural Language Processing (NLP) tasks are performed on the provided texts in order to automatically extract the most significant features. We observed that most key phrases in patent texts are noun phrases. Therefore, we built a large-scale, efficient, and distributed NLP pipeline for extracting noun terms and phrases for each sentence in provided texts. This allows us to reduce the overall amounts of relevant terms, and helps us to identify core technological terms to assist in further analyses. The NLP pipeline consists of a sequence of steps, where each stage performs a specific processor task on the regarded part of text such as sentence detection, tokenization, part-of-speech tagging, shallow syntactic parsing, stopwords removal, stemming, and pruning.

Analysis of Technological Areas. Patent classification schemes such as the IPC are assigned by experts to classify the patent documents according to their technology area. Many studies have been conducted in order to obtain the features or trends of technology development by using patent search and the IPC[10] codes. However, these classifications have proved to be quite inconsistent in various aspects due to their complexity on the one hand and the lack of coverage for all areas of technology on other hand, in particular, for the emerging technology areas. We have observed that most of trending areas in the computer science

[8] https://bitbucket.org/wwmm/oscar4.
[9] https://www.ebi.ac.uk/unichem/.
[10] http://www.wipo.int/classifications/ipc/en/.

Table 1. Examples of 'Technical Field' texts.

Patent number	Text of technical field
PCT2016015041	The present invention generally relates to blockchain technology. Specifically, this invention relates to creating a blockchain called a slidechain that allows for multiple valid branches or forks to propagate simultaneously with a customized set of protocol rules embedded in and applied to each fork chain that branches from another chain
EP2013068285	The present invention belongs to the field of the photonic Internet of Things technologies, and specifically, to a visible-light communication-based encryption, decryption and encryption/decryption method and system

domain such as Deep Learning, Big Data Cloud, Blockchain, and Internet of Things (IoT) are missing in the descriptions of the patent classification scheme, but they clearly appear in texts of the technical field segment as the examples in Table 1 show.

As a consequence, we analyzed the patent texts in order to extract implicit or unknown technology areas of patents from the technical field section of a patent, which is a brief paragraph, typically one or two sentences long, describing the accurate technological areas to which the invention relates to. The idea is to use the technological information from the technical field instead of the entire text in order to discover trending technical areas for specific topics in order to better understand the inventions and their associated technological fields.

Figure 2 shows KNIME workflows that are composed from different custom nodes for searching, analyzing, and visualizing large-scale patent data. We extracted 26388 patents that are related to the field of "Information Security" from two big patent databases, Patent Cooperation Treaty (PCTFULL) and European Patents (EPFULL) from our patent repository. 18381 of the extracted patents had detailed description texts. Running the PatSeg tool to extract the text of the technical field segment, we obtained 12476 patents having the technical field section. By using another workflow (see Fig. 2), we run our Spark-NLP service for extracting the significant technological terms and phrases, calculating their frequencies, and exporting the results automatically for further analyses.

As each technical term is treated as a topic and the relationships between terms are simply based on co-occurrence statistics, they can indirectly represent the semantic information of the related terms. A co-occurrence map served to describe the technology hotspots of the "Information Security" area at a lower level. Based on the frequency of terms, we collected the top 200 terms and visualized them via VOSviewer [22]. Figure 3 provides the main technology hotspots in the area of information security such as digital signature, public key cryptography, private key cryptography, access control, authentication systems, cryptographic algorithms, and key management. It also shows more accurate areas such as hash function, DRM systems (Digital Rights Management), and security with smart card.

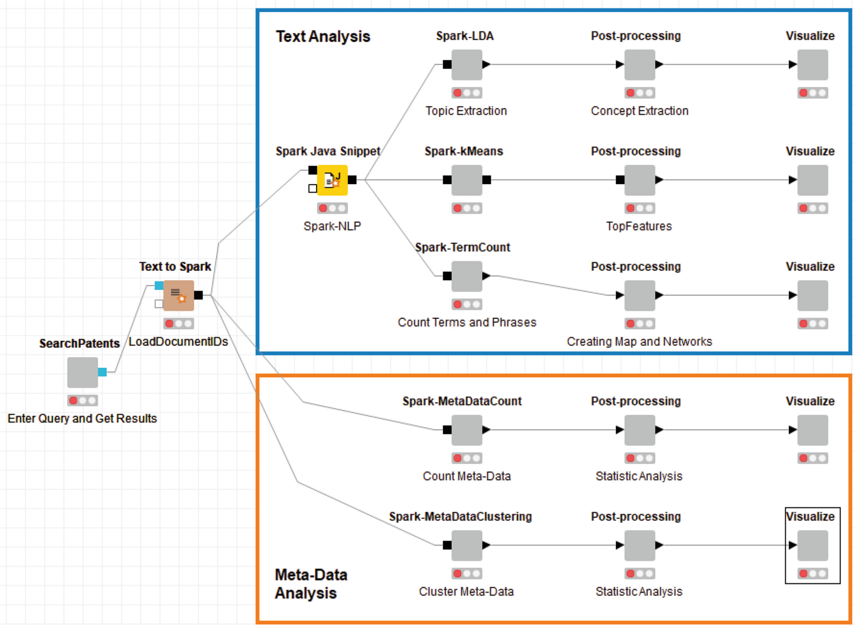


Fig. 2. Workflows for spark-based patent analytics in KNIME.

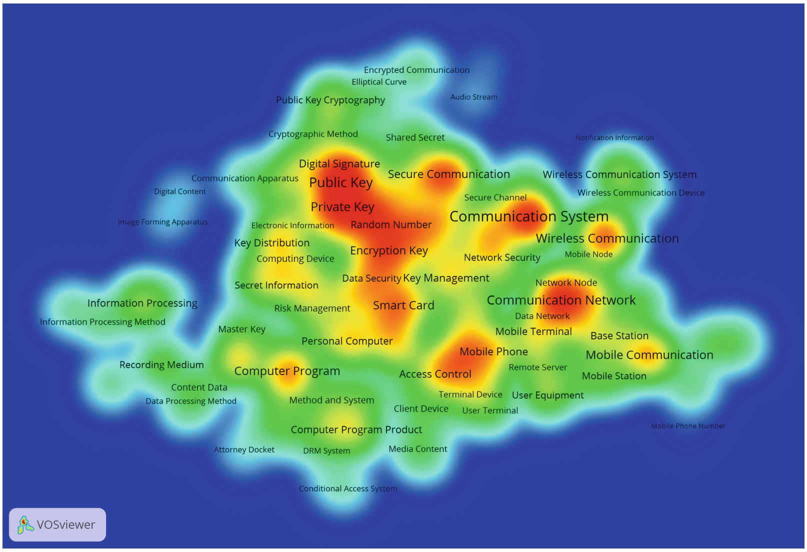


Fig. 3. Co-occurrence Map for ‘Information Security’.

Technological Concept Extraction. Patents tend to contain very specialized terminologies and technology-specific paraphrasing. So we need to go deeper in order to discover the basic knowledge constituting the subject matter of the invention. Therefore, in this analysis we extracted the topics and technology concepts. Latent Dirichlet Allocation (LDA) is an analytical method that can be used to detect topics from texts and extract keywords corresponding to each topic, since the relation between topics and keywords can be derived using the probability of the keywords for each topic based on the Dirichlet probability distribution.

LDA has been used as a method to assist the topic discovery and analysis of the patent texts [23–25]. In order to discover the technology concepts in the patent domain, we used LDA to extract the list of topics in a specific collection of patent documents by employing a KNIME workflow (Fig. 2). For each topic, we obtained a set of candidate concepts (noun terms and phrases) which represent that topic, where these candidate concepts have top scores in the related topics. The most active concepts are extracted as follows: each candidate concept gets a score depending on its frequency and the average likelihood of occurrence in the patent corpus. Then, the candidate concepts of all topics are ranked and the top concepts with the highest score are returned. For our analysis, we selected the "Information Security" topic and retrieved all patents related to that topic retrieving 26261 documents published between 1978 and 2015. Figure 4 shows the publication trend chart for "Information Security" applying a linear fit approach to estimate the future trend.

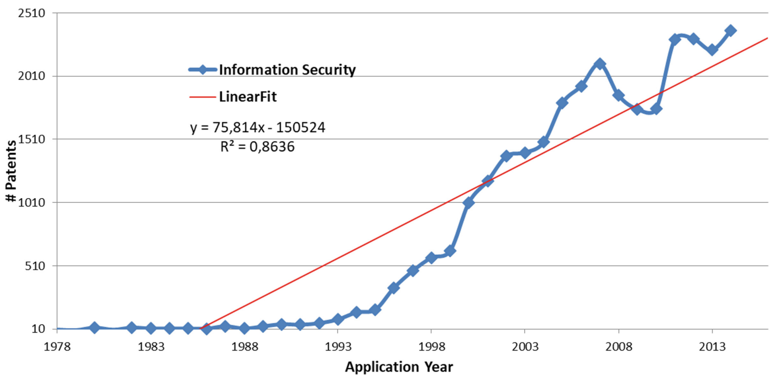


Fig. 4. The publication trend of the 'Information Security'.

In practice, we used our annotation services in order to extract the text of most important segments from each patent such as title, abstract, technical field, summary, and independent claims. Our Spark-NLP pipeline is used to process the texts and extract only the important nouns and noun phrases. For the phrases that consist of more than two words in length, we calculate n-grams. We have built a Spark machine learning pipeline to apply the Spark-LDA algorithm. The sequence starts with tokenized noun phrases by linking all words of a noun phrase via short lines and treating them as single words in order to be represented as a token. Herewith, a collection of text documents is converted to vectors of token counts.

Finally, the Spark-LDA estimator is fit on the vectors to produce the model. The two hyper-parameters α, and β controlling the amount of smoothing are applied to the topic distributions for each patent document and the word distributions for each topic, set as 0.4 and 0.02, respectively; and the number of iterations of Gibbs sampling was set to 200. Then, we applied our methods of concept extraction on the results of the LDA algorithm.

Knowledge of emerging concepts on the other hand is particularly important to patent experts and companies who are interested in monitoring a particular field or business. For example, a "touch devices" company wants to review the new technologies, products, and materials that have been produced in the market only in last five years, in order to make a decision for developing a new product. Consequently, our method can be used to extract the emerging concepts for a specific date span, and are described as follows: extract all concepts as we have done it previously, define the time span (five years), each extracted concept will be selected as a new concept if: it occurs in patents with an application date in a selected time span, and does not appear in a patent with an application date before this time span.

The above analysis answered the question "What is the information security community interested in?". And in order to answer the question: "How does such technical concept changes over the time?", we used regression to analyze the trends. Particularly, based on the document frequency of the extracted concepts, we draw the trends of the hot concepts. Figure 5 shows mainly two types of trends: upward trend (Authentication Systems) with a positive slope where the number of related patent is increasing significantly by year, and downward trend

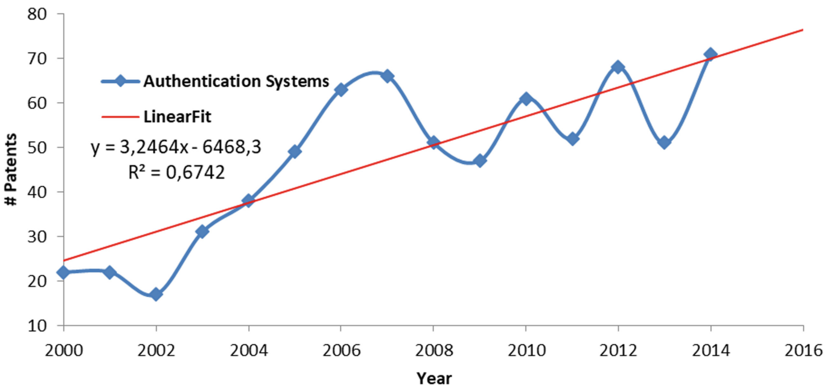


Fig. 5. Trend forecasting of concept 'Authentication Systems'.

(Digital Signature) with a negative slope. Moreover, a prediction model was developed by calculating the frequency of related concepts in all patents using the data from year 2000 to 2014 for the target topic and applying a linear regression model to estimate the future trends.

4.3 Performance Evaluation

This section illustrates the experiments designed to perform our services on big data of patents. We conducted our experiments on a cluster based on the Hortonworks Data Platform, which consists of two namenodes and 14 datanodes. Each namenode is DELL PowerEdge R730 with 2×8 Core Intel(R) Xeon(R) CPU 2.40 GHz and 128 GB of RAM, and each datanode is DELL PowerEdge R430 with 2×8 Core Intel(R) Xeon(R) CPU 2.40 GHz and 192 GB of RAM and about 45 TB space for HDFS with CentOS7 operating system.

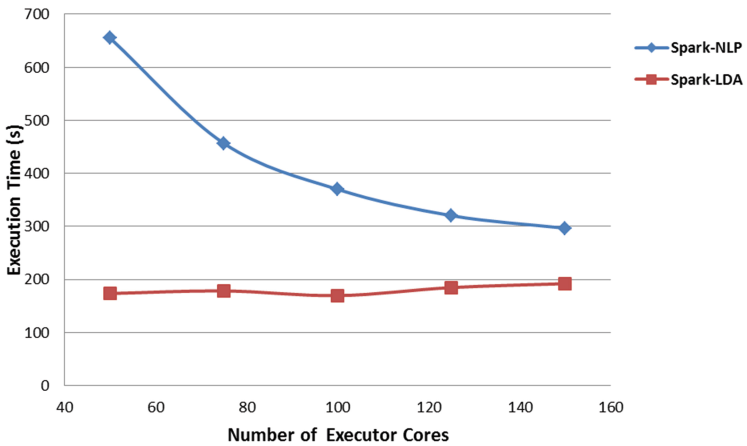


Fig. 6. Execution performance of spark services.

Apache Hadoop MapReduce 2.7 and Apache Spark 1.6 are used to implement the services. Figure 6 shows the performance and efficiency of Spark-based services for analytics employing a NLP pipeline and LDA. Spark-NLP and Spark-LDA services are run on a dataset for the "Information Security" topic area (26388 patents). For each service, we set 100 GB of RAM for the Spark driver, 25 GB of RAM for each executor since each executor has only five cores. The number of partitions is set to 100. We set additional parameters for Spark-LDA; the number of topics as 30, number of iterations as 100, and the hyper-parameters α, and β were set as 0.01 and 0.05 respectively.

We also used a serialized RDD and Spark's KryoSerializer to decrease the overall memory usage and GC overhead. Figure 6 shows the execution performance of Spark-NLP and Spark-LDA services with varying number of cores and 26388 patent documents. By increasing the number of executors, the execution time of Spark-NLP decreased significantly, and the execution time of Spark-LDA including the time of creating text vectors did not improve much with increasing number of executors, because the implementation of LDA in Spark's MLlib library contains a large amount of shuffling, a process that requires data to be transferred from every executor to the driver to be totaled and then redistributed in the next iteration of LDA. Moreover, increasing the number of partitions will drastically increase the number of network communications and disk I/O. This becomes a bottleneck in LDA Spark jobs, as they rely heavily on memory for speed, and transferring data over the network or to disk is much slower than keeping it in memory. Therefore, tuning parameters of Spark job should consider the characteristics of resources in the cluster, and size, serialization and persisting the data, as well as the number of partitions and shuffling.

5 Conclusion

The work described in this paper presented a scalable patent mining and analytics framework built on top of a big-data architecture and a scientific workflow system for allowing the user to efficiently annotate, analyze and interact with patent data on a large-scale via visual interaction. The architecture of the framework was presented and different scalable services for the semantic enrichment of a single patent, and analysis of large collections of patent documents were described. In two case studies, we have shown how this framework can be utilized to search, analyze and interact with patent information, and create reasonable analytics workflows by interconnecting scalable services for patent analysis in form of nodes in a visual workflow system. Our future directions of research will continue to improve the semantic enrichment and analysis pipeline of patent data by using intelligent algorithms such as word embeddings and Deep Learning, ontological knowledge and linked open data as well as making use of cloud services in order to improve the execution performance and scalability of the patent mining services.

References

1. Hong, S.: The Magic of Patent Information. http://www.wipo.int/sme/en/documents/patent_information_fulltext.html
2. Yoon, J., Kim, K.: TrendPerceptor: a property function based technology intelligence system for identifying technology trends from patents. Expert Syst. Appl. **39**(3), 2927–2938 (2012)
3. Choi, S., Park, H., Kang, D., Lee, J.Y., Kim, K.: An SAO based text mining approach to building a technology tree for technology planning. Expert. Syst. Appl. **39**(13), 11443–11455 (2012)
4. Trappey, C.V., Wu, H.Y., Taghaboni-Dutta, F., Trappey, A.J.C.: Using patent data for technology forecasting: China RFID patent analysis. Adv. Eng. Inform. **25**(1), 53–64 (2011)
5. Daim, T.U., Gomez, F.A., Martin, H., Sheikh, N.: Technology roadmap development process (TRDP) in the medical electronic device industry. Int. J. Bus. Innov. Res. **7**(2), 228–263 (2013)

6. Lee, Y., Kim, S., Shin, J.: Technology opportunity identification customized to the technological capability of SMEs through two-stage patent analysis. Scientometrics **100**(1), 227–244 (2014)
7. Abbas, A., Zhang, L., Khan, S.U.: A literature review on the state-of-the-art in patent analysis. World Pat. Inf. **37**, 3–13 (2014)
8. Hu, J., Li, S., Yao, Y., Yu, L., Yang, G., Hu, J.: Patent keyword extraction algorithm based on distributed representation for patent classification. Entropy **20**, 104 (2018)
9. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proceedings of the 27th Annual Conference on Neural Information Processing Systems (NIPS), Lake Tahoe, NV, USA (2013)
10. Beltz, H., Fueloep, A., Wadhwa, R.R., Erdi, P.: From ranking and clustering of evolving networks to patent citation analysis. In: 2017 International Joint Conference on Neural Networks (IJCNN), vol. 350. IEEE (2017)
11. Jun, S., Park, S.-S., Jang, D.-S.: Document clustering method using dimension reduction and support vector clustering to overcome sparseness. Expert. Syst. Appl. **41**(7), 3204–3212 (2014)
12. Du, R., Drake, B., Park, H.: Hybrid clustering based on content and connection structure using joint nonnegative matrix factorization, arXiv preprint arXiv:1703.09646
13. Seo, W., Kim, N., Choi, S.: Big data framework for analyzing patents to support strategic R&D planning (2016)
14. Dean, J., Ghemawat, S.: MapReduce: simplified data processing on large clusters. In: Proceedings of the 6th Conference on Symposium on Operating Systems Design & Implementation, OSDI 2004 (2004)
15. Tseng, Y., Lin, C., Lin, Y.: Text mining techniques for patent analysis. Inf. Process. Manag. **43**(5), 1216–1247 (2007)
16. Sofean, M.: Automatic segmentation of big data of patent texts. In: Bellatreche, L., Chakravarthy, S. (eds.) DaWaK 2017. LNCS, vol. 10440, pp. 343–351. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-64283-3_25
17. Hackl-Sommer, R., Schwantner, M.: Patent claim structure recognition. Arch. Data Sci. Ser. A **2**(1), 15 (2017)
18. Aras, H., Hackl-Sommer, R., Schwantner, M., Sofean, M.: Applications and challenges of text mining with patents. In: IPaMin@KONVENS (2014)
19. Vazquez, M., Krallinger, M., Leitner, F., Valencia, A.: Text mining for drugs and chemical compounds: methods, tools and applications. Mol. Inform. **30**, 506–519 (2011)
20. Matos, P., Alcaentara, R., Dekker, A., Ennis, M., Steinbeck, C.: Chemical entities of biological interest: an update. Nucleic Acids Res. **38**, D249–D254 (2010)
21. Trippe, A.: Guidelines for Preparing Patent Landscape Reports. Patinformatics, LLC, With contributions from WIPO Secretariat (2015)
22. Waltman, L., van Eck, N.J., Noyons, E.C.: A unified approach to mapping and clustering of bibliometric networks. J. Inform. **4**(4), 629–635 (2010)
23. Tang, J., et al.: PatentMiner: topic-driven patent analysis and mining. In: KDD 2012 (2012)
24. Ankam, S., Dou, W., Strumsky, D., Zadrozny, W.: Exploring emerging technologies using patent data and patent classification. In: CHI 2012 (2012)
25. Chen, H., Zhang, Y., Zhang, G., Zhu, D., Lu, J.: Modeling technological topic changes in patent claims. In: Proceedings of PIC MET 2015 (2015)

Software Engineering: Special Session on Intelligent Systems and Software Engineering Advances

SVM Accuracy and Training Speed Trade-Off in Sentiment Analysis Tasks

Konstantinas Korovkinas[1(✉)], Paulius Danėnas[2], and Gintautas Garšva[1]

[1] Kaunas Faculty, Vilnius University, Muitines str. 8, 44280 Kaunas, Lithuania
konstantinas.korovkinas@knf.vu.lt, gintautas4garsva@gmail.com
[2] Center of Information Systems Design Technologies,
Department of Information Systems, Kaunas University of Technology,
Studentu Str. 50-313a, Kaunas, Lithuania
paulius.danenas@ktu.lt

Abstract. SVM technique is one of the best techniques to classify data, but it has a slow performance in the big data arrays. This paper introduces the method to improve the speed of SVM classification in sentiment analysis by reducing the training set. The method was tested on the Stanford Twitter sentiment corpus dataset and Amazon customer reviews dataset. The results show that the execution time of the introduced method outperforms the standard SVM classification method.

Keywords: SVM · Big data arrays · Sentiment analysis

1 Introduction

Sentiment analysis became very popular since people started using Facebook, Twitter, Instagram and other social networks. The main goal of research related to sentiment analysis is to obtain authors' feelings, expressed in positive or negative comments [11].

Authors in [6,9,10,23] compared classifiers Naïve Bayes, maximum entropy, and support vector machines for sentiment analysis and concluded that SVM proved to perform best. Chorowski in [2] compared four classifier models, i.e., the Support Vector Machine (SVM), the LeastSquares SVM (LSSVM), the Extreme Learning Machine (ELM), and the Margin Loss ELM (MLELM). The comparison revealed that ELMs based classifiers outperform SVMs in terms of training speed. However, the accuracy results were slightly lower for ELMs than for SVMs. Le and Nguyen in [12] proposed a sentiment analysis model based on Naïve Bayes and Support Vector Machine in purpose to analyze sentiment more effectively. Korovkinas et al. in [11] used the SVM and Naïve Bayes for creation of ensemble method and stated that these methods are still prominent for future research and can be used to develop new hybrid techniques based on their combination. Manek et al. in [17] proposed a Gini Index based feature selection method with Support Vector Machine (SVM) classifier for sentiment

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 227–239, 2018.
https://doi.org/10.1007/978-3-319-99972-2_18

classification on a large movie review dataset. Their method provided better classification performance in terms of reduced error rate and accuracy.

According to a number of authors, who worked on SVM hyperparameter optimization, SVM proved its efficiency to solve difficult tasks in various domains. Damaševičius in [4] used SVM for classification of DNA sequences and recognition of the regulatory sequences. The results demonstrated that selection of the kernel type and its parameters have direct impact on the performance of the SVM and accuracy of the results. Sunkad in [26] proposed the best set of features and the SVM hyperparameters for obtaining the best results in human activity recognition. Osman et al. in [22] tuned hyperparameters of two machine learning algorithms to improve bug prediction accuracy. They concluded that the k-nearest neighbours algorithm always significantly improved and the prediction accuracy of support vector machines either improved or was at least retained. Liu and Zio in [16] used one synthetic dataset and two real time series data, related to prediction of wind speed in a region and leakage from the reactor coolant pump in a nuclear power plant and proposed the preferable choice for tuning SVM hyperparameters for recursive multi-step-ahead prediction.

However, despite all advantages, typical for SVM algorithm, it is characterized by slow performance in the big data arrays. The higher number of features is, the longer computation time it requires. There have been a number of efforts to speed up SVM, and most of them focus on reduction of the training set. Lee and Mangasarian in [13] proposed the Reduced Support Vector Machine (RSVM) algorithm which uses a randomly selected subset of the data that is typically 10% or less of the original dataset to obtain a nonlinear separating surface. RSVM performs better than a conventional SVM, sequential minimal optimization (SMO) and projected conjugate gradient chunking (PCGC). Lei and Govindaraju in [14] introduced a reduction of the feature space using principal component analysis (PCS) and Recursive Feature Elimination (RFE). PCA and RFE can speed up the evaluation of SVM by an order of 10 while maintaining comparable accuracy. Graf et al. in [8] proposed the Cascade SVM, where the training set is first divided into a number of subsets and then these subsets are optimized by multiple SVMs. The partial results were combined and filtered again in the "Cascade" of SVMs, until the global optimum was reached. Later, Meyer et al. in [19] introduced a new stepwise bagging approach that exploits parallelization in a better way than the Cascade SVM and contains an adaptive stopping-time to select the number of stages for improving accuracy. Nandan et al. in [21] used a linear time algorithm based on convex hulls and extreme points to select subset, the so-called representative set of the training data for SVM optimization. Wang et al. in [28] reduced SVM training time using only the most informative samples, obtained after removing most of the training data. Guo and Boukir in [7] proposed a new ensemble margin-based data selection approach based on a simple and efficient heuristic to provide support vector candidates: they selected the lowest margin instances that reduced SVM training task complexity while maintaining the accuracy of the SVM classification. Mao et al. in [18] trained number of kernel SVMs on the randomly selected small subsets of training data and concluded that it is more efficient than training a single kernel SVM on the whole training data especially for large datasets. Mourad et al. in [20] proposed a computationally efficient subset selection algorithm for fast SVM training on large scale data. Liu et al. in [15] proposed to train an approximate SVM by using the anchors obtained from non-negative matrix factorization (NMF) in a divide-and-conquer framework.

The results led to the conclusion that SVM is the best choice for our hybrid sentiment recognition system which will be introduced in the future works. This paper presents a part of the aforementioned system: a method to improve SVM classification speed by reducing a training set and to test it on the datasets, containing a large volume of data – big data arrays, in order to recognize positive or negative sentiments and to compare the execution time and accuracy. The rest of paper is organized as follows. Section 2 defines SVM which was used in the experiment and our proposed method. Section 3 gives a description of datasets, used in experiments; experiments; experimental settings; calculation of effectiveness and results. Section 4 presents the conclusions and tasks for future works.

2 Methodology

This section introduces our proposed method and presents support vector machines algorithm used in it.

2.1 Support Vector Machines

Support vector machines were introduced in [1,3] and basically attempt to find the best possible surface to separate positive and negative training samples in supervised manner.

In this subsection we describe linear SVM (Fan et al. in [5]) which is optimized for large-scale learning and is used in this paper.

Given training vectors $x_i \in R^n$, $i = 1, \dots, l$ in two class, and a vector $y \in R^l$ such that $y_i = \{1, -1\}$, a linear classifier generates a weight vector w as the model. The decision function is

$$sgn(w^T x)$$

$L2$-regularized $L1$-loss SVC solves the following primal problem:

$$\min_w \frac{1}{2} w^T w + C \sum_{i=1}^{l} (\max(0, 1 - y_i w^T x_i))$$

whereas L2-regularized L2-loss SVC solves the following primal problem:

$$\min_w \frac{1}{2} w^T w + C \sum_{i=1}^{l} (\max(0, 1 - y_i w^T x_i))^2 \tag{1}$$

Their dual forms are:

$$\min_{\alpha} \frac{1}{2}\alpha^T \bar{Q}\alpha - e^T\alpha$$

$$\text{subject to } \; 0 \leq \alpha_i \leq U, \;\; i = 1, \ldots, l$$

where e is the vector of all ones, $\bar{Q} = Q + D$, D is a diagonal matrix, and $Q_{ij} = y_i y_j x_i^T x_j$. For $L1$-loss SVC, $U = C$ and $D_{ii} = 0, \forall i$. For $L2$-loss SVC, $U = \infty$ and $D_{ii} = 1/(2C), \forall i$.

$L1$ regularization generates a sparse solution w. $L1$-regularized $L2$-loss SVC solves the following primal problem:

$$\min_{w} \|w\|_1 + C\sum_{i=1}^{l}(\max(0, 1 - y_i w^T x_i))^2 \tag{2}$$

where $\|\cdot\|_1$ denotes the 1-norm (Fan et al. in [5]).

2.2 Proposed Method

The proposed method is based on selection of the training data size subject to the subset of split testing data. Thus, the testing data is split into equal subsets and training data size is calculated on the basis of the size of the first subset. It is assumed that the testing subset is 30%, therefore, the training data should be 70%. "Results" is the final results set with the following classified sentiment: "positive" or "negative". The diagram and algorithm of the proposed method are presented below (Fig. 1).

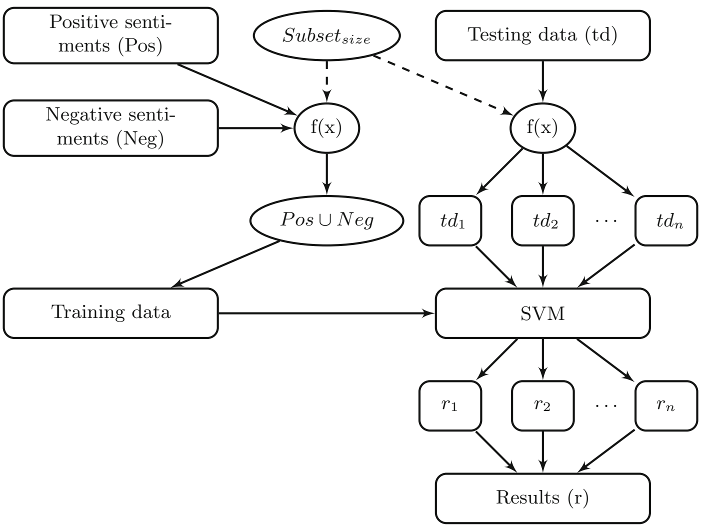


Fig. 1. Proposed method

Algorithm

Input: Pos_{data} – set of positive sentiments for training;
Neg_{data} – set of negative sentiments for training;
td – set of testing data subsets;
$Subset_{size}$ – size of testing data subset is divided into;
D_{train} – set of training data;
D_{test} – set of testing data;
$Train_{count}$ – count of sentiments should be selected from Pos_{data} and Neg_{data} sets. This value is calculated by formula:
$Train_{count} = ((Subset_{size}/2) * (train_{size}/test_{size})))).$ $Subset_{size}$ is divided by 2, because we need to select equal parts from Pos_{data} and Neg_{data} sets;
POS_{train} – set of randomly selected sentiments from Pos_{data} set;
NEG_{train} – set of randomly selected sentiments from Neg_{data} set;
$R_{SVM} = \{SVMsent\}$ – set of SVM results, $SVMsent$ – sentiment;

$R_{SVM} = \{\}$
$D_{train} = \{\}$
$POS_{train} = (random.sample(Pos_{data}, Train_{count}))$
$NEG_{train} = (random.sample(Neg_{data}, Train_{count}))$
$D_{train} = POS_{train} \cup NEG_{train}$
train SVM with D_{train}
n = 0;
for i = 1 : $trunc(len(D_{test})/Subset_{size})$
    $td_i = D_{test}[(n+1) : (Subset_{size} * i),]$
    pass td_i *to SVM input (output* $SVMsent_i$*)*
    $R_{SVM} = R_{SVM} \cup \{SVMsent_i\}$
    $n = (Subset_{size} * i)$
if $(len(D_{test})\ \%\ Subset_{size}) > 0$
    $td_{i+1} = D_{test}[(n+1) : (len(D_{test}),]$
    pass td_{i+1} *to SVM input (output* $SVMsent_{i+1}$*)*
    $R_{SVM} = R_{SVM} \cup \{SVMsent_{i+1}\}$

Output: set of classification results $Results = \{sentiment\}$

3 Experiments and Results

3.1 Dataset

In this paper are used two existing datasets: the Stanford Twitter sentiment corpus dataset[1] and Amazon customer reviews dataset[2]. The Stanford Twitter sentiment corpus dataset is introduced by Go et al. in [6] and contains 1.6 million tweets automatically labeled as positive or negative based on emotions. The dataset is splitted in training dataset (560K positive and 560K negative

[1] http://help.sentiment140.com/.
[2] https://www.kaggle.com/bittlingmayer/amazonreviews/.

tweets, in total 1.12M tweets) and testing dataset (480K tweets). Amazon customer reviews dataset contains 4 million reviews and star ratings. The dataset is splitted into training dataset (1.4M positive and 1.4M negative reviews, in total 2.8M reviews) and testing dataset (1.2M reviews).

Training and testing data has been preprocessed and has been cleaned before it was passed as the input of SVM algorithm. It included removing redundant tokens such as hashtag symbols @, numbers, http for links, punctuation symbols, etc. After cleaning was performed all datasets were checked and empty strings were removed.

3.2 Experiments

Ten experiments are performed in this paper: five experiments with the Stanford Twitter sentiment corpus dataset (sentiment140) and five experiments with Amazon customer reviews dataset (Amazon reviews).

Original linear SVM technique is used in the first and the second experiments, using typical split into 70% for training and 30% for testing, and applying it to the Stanford Twitter sentiment corpus dataset and to Amazon customer reviews dataset. The main goal of the aforementioned experiments is to compare the efficiency of an ordinary SVM technique and our proposed method, which is applied in the further experiments of this paper.

In the third and seventh experiments, the testing data is used from the first (480K tweets) and second experiments (1.2M reviews). Furthemore, it is divided into subsets, which contain 30K rows of a dataset. The last subset contains the remainder after division, if the testing dataset cannot be divided into equal subsets without the remainder. Similar splitting into subsets is used in other experiments, using different size of subsets: 60K rows of a dataset are used in the fourth and eight experiments; in the fifth and ninth experiments – 120K rows of a dataset; in the sixth and tenth experiments – 180K rows of a dataset.

It is important to note that all experiments are performed 20 times to get more accurate results; the MIN, the MAX and the average are taken as the final results. Training data is calculated subject to the subset of testing data as it is described in the proposed method (see Subsect. 2.2) and new training set is randomly selected for each experiment and for each execution time.

Detailed experimental settings are presented in Tables 1 and 2 (see Subsect. 3.3).

3.3 Experimental Settings

Data cleaning and preparing are performed with R [27]. The experiments are implemented with Python programming language and scikit-learn [24]: library for machine learning. Tables below show initial settings for SVM input.

We used LinearSVC module for SVM classification with this default parameters (all parameters are selected as they are in LinearSVC module). It is similar to SVC (implementation of conventional SVM) with parameter kernel = "linear",

Table 1. Ordinary SVM technique, using typical dataset split

Exp. No.	Dataset	Training data 70%	Testing data 30%
1	sentiment140	1.12M	480K
2	Amazon reviews	2.8M	1.2M

Table 2. Experimental settings for proposed method

Exp. No.	Dataset	Testing data size (TDs)	Subset size (SubS)	Subsets quantity (SQ) trunc (TDs/Ss)	Remainder TDs-(SubS*SQ)	Calculated training data dependently on SubS
3	sentiment140	480K	30K	16	0	70K
4		480K	60K	8	0	140K
5		480K	120K	4	0	280K
6		480K	180K	2	120K	420K
7	Amazon reviews	1.2M	30K	40	0	70K
8		1.2M	60K	20	0	140K
9		1.2M	120K	10	0	280K
10		1.2M	180K	6	120K	420K

but implemented in terms of LibLinear (A Library for Large Linear Classification[3]) rather than LibSVM (A Library for Support Vector Machines[4]), so it has more flexibility in the choice of penalties and loss functions and should scale better to large numbers of samples [24]. The main goal of this research is to compare training speed between our method and ordinary SVM on equal terms, consequently we don't need to change values of SVM parameters, cause it is enough to compare accuracy of methods obtained with default parameters.

Support Vector Machines parameters:

- kernel: linear.
- C (Penalty parameter of the error term. It is the only parameter for linear classification.). Type: float, optional (default = 1.0) [24].

For experiments is used computer with processor Intel(R) Core(TM) i7-4712MQ CPU @ 2.30 GHz and 8.00 GB installed memory (RAM).

[3] https://www.csie.ntu.edu.tw/~cjlin/liblinear/.
[4] https://www.csie.ntu.edu.tw/~cjlin/libsvm/.

3.4 Effectiveness

SVM classification time is measured with seconds (sec). Effectiveness is measured using statistical measures: accuracy, precision, recall and $F_1 score$. Formulas are presented below (Sammut and Webb in [25]):

Accuracy: $ACC = \dfrac{TP + TN}{TP + TN + FP + FN}$

Precision. Positive predictive value: $PPV = \dfrac{TP}{TP + FP}$

Precision. Negative predictive value: $NPV = \dfrac{TN}{TN + FN}$

Recall. True positive rate: $TPR = \dfrac{TP}{TP + FN}$

Recall. True negative rate: $TNR = \dfrac{TN}{TN + FP}$

Harmonic mean of PPV and TPR: $F_1 score = \dfrac{2}{\frac{1}{PPV} + \frac{1}{TPR}}$

where TP – count of correctly classified "positive" sentiments, TN – count of correctly classified "negative" sentiments. FP – count of incorrectly classified "positive" sentiments. FN – count of incorrectly classified "negative" sentiments.

Calculation is performed with scikit-learn *metric* module [24].

3.5 Results

The results of an ordinary SVM technique (see Table 3) revealed that in case of sentiment140 dataset the execution time average for is 407,26 s, meanwhile the accuracy (ACC) average is 79,10%. While referring to Amazon customer reviews dataset, the execution time average is 1031,64 s and the accuracy (ACC) average is 89,44%.

Table 3. Results of ordinary SVM technique, using typical dataset split

Dataset	Function	Time (sec)	ACC	PPV	NPV	TPR	TNR	$F_1 score$
sentiment140	MIN	372,79	79,09%	78,81%	79,31%	79,46%	78,59%	79,17%
	MAX	464,71	79,11%	78,89%	79,38%	79,59%	78,72%	79,21%
	AVG	407,26	79,10%	78,84%	79,36%	79,56%	78,64%	79,20%
Amazon reviews	MIN	1004,93	89,30%	91,49%	86,74%	85,81%	91,89%	88,91%
	MAX	1053,14	89,62%	92,31%	87,85%	87,28%	92,85%	89,37%
	AVG	1031,64	89,44%	91,97%	87,21%	86,44%	92,45%	89,11%

Table 4 shows the results of the proposed method for the sentiment140 dataset. Average execution time of our method ranges from 7,47 s to 51,44 s Therefore, it outperforms an ordinary SVM technique with the average of 407,26 s; however, the accuracy (ACC) average of our method ranges from

Table 4. Results of the proposed method applied on the Stanford Twitter sentiment corpus dataset

Subset	Function	Time (sec)	ACC	PPV	NPV	TPR	TNR	$F_1 score$
30K	MIN	7,39	76,72%	76,44%	76,92%	76,96%	76,17%	76,85%
	MAX	7,67	77,03%	76,89%	77,39%	77,75%	76,86%	77,16%
	AVG	7,47	76,87%	76,66%	77,10%	77,30%	76,45%	76,98%
60K	MIN	10,06	77,52%	77,21%	77,70%	77,86%	76,89%	77,59%
	MAX	10,86	77,69%	77,54%	78,00%	78,30%	77,42%	77,79%
	AVG	10,19	77,61%	77,36%	77,86%	78,08%	77,14%	77,72%
120K	MIN	21,27	78,08%	77,77%	78,37%	78,49%	77,51%	78,20%
	MAX	25,78	78,35%	78,13%	78,65%	78,91%	77,96%	78,46%
	AVG	23,28	78,23%	77,97%	78,50%	78,70%	77,76%	78,34%
180K	MIN	47,73	78,47%	78,22%	78,69%	78,86%	77,96%	78,56%
	MAX	55,28	78,62%	78,37%	78,95%	79,20%	78,21%	78,74%
	AVG	51,44	78,55%	78,29%	78,82%	79,02%	78,08%	78,65%

76,87% to 78,55% and is slightly lower than the ordinary SVM technique (ACC 79,10%). Moreover, the results clearly demonstrate that the performance in terms of *PPV, NPV, TPR, TNR,* $F_1 score$ is very similar in all experiments.

Table 5 presents the results for Amazon customer reviews dataset. From here, it is evident that the proposed method again shows very good execution time: the average from 67,17 s to 105,08 s while it took 1031,64 s for ordinary SVM. The accuracy (ACC) average from 87,63% to 89,09% was slightly smaller than in case

Table 5. Results of the proposed method applied on Amazon customer reviews dataset

Subset	Function	Time (sec)	ACC	PPV	NPV	TPR	TNR	$F_1 score$
30K	MIN	65,41	87,55%	87,55%	87,33%	87,24%	87,54%	87,55%
	MAX	70,02	87,70%	87,91%	87,71%	87,73%	88,01%	87,68%
	AVG	67,17	87,63%	87,70%	87,57%	87,55%	87,72%	87,62%
60K	MIN	71,67	88,21%	88,19%	88,16%	88,14%	88,17%	88,20%
	MAX	75,24	88,34%	88,39%	88,37%	88,39%	88,42%	88,33%
	AVG	73,73	88,27%	88,28%	88,25%	88,25%	88,29%	88,27%
120K	MIN	85,31	88,79%	88,78%	88,73%	88,71%	88,77%	88,79%
	MAX	97,50	88,86%	88,93%	88,86%	88,87%	88,96%	88,85%
	AVG	88,54	88,82%	88,85%	88,79%	88,79%	88,85%	88,81%
180K	MIN	102,96	89,06%	89,00%	88,99%	88,98%	88,99%	89,06%
	MAX	109,63	89,12%	89,17%	89,12%	89,14%	89,19%	89,12%
	AVG	105,08	89,09%	89,11%	89,07%	89,06%	89,11%	89,08%

of an ordinary SVM method with the average of 89,44%. The performance in terms of *PPV, NPV, TPR, TNR,* F_1score is almost identical in all experiments.

Figures 2 and 3 graphically depict the results of the accuracy and F_1score. In Figs. 4 and 5 – the results of *PPV, NPV, TPR, TNR*. When Amazon customer reviews dataset was used, our method performed better with higher *NPV* and *TPR* (see Fig. 5).

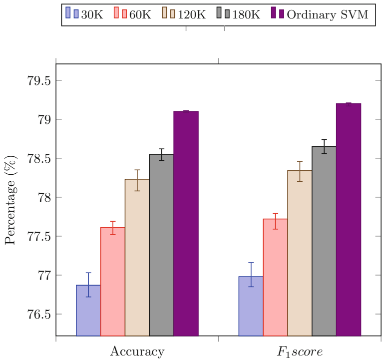


Fig. 2. sentiment140 results

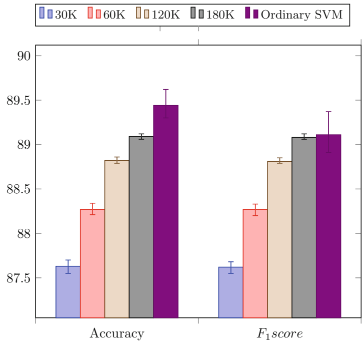


Fig. 3. Amazon reviews results

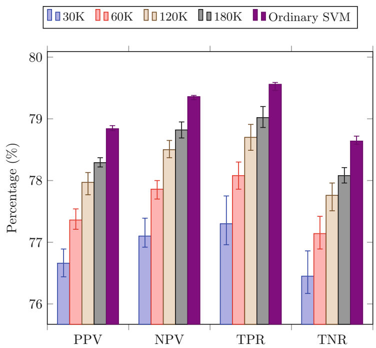


Fig. 4. sentiment140 results

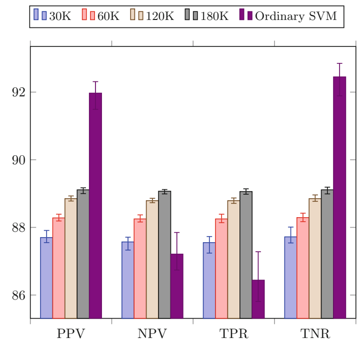


Fig. 5. Amazon reviews results

Figure 6 compares the execution time between an ordinary SVM technique and our method, when the Stanford Twitter sentiment corpus and Amazon customer reviews datasets are used. It is important to conclude that the proposed technique outperformed an ordinary SVM technique in all cases.

The achieved results are not objective in comparison with other works as different hardware and methods (if implementations are not presented), datasets, parameters, tasks, etc. are used.

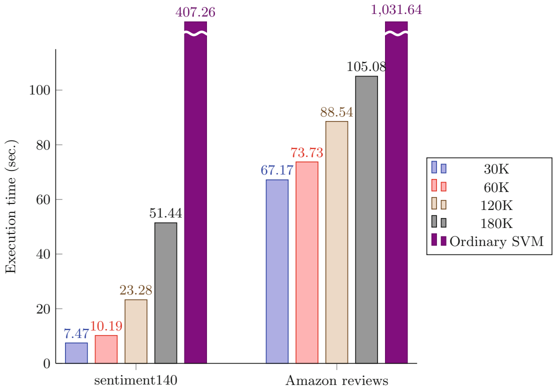


Fig. 6. Execution time comparison

4 Conclusions and Future Work

Training data size has significant impact on SVM classification speed. Properly selected features can improve executing time with no losing or similar accuracy, like shown in papers [7,14,21] etc. This paper proposes the method to improve SVM classification speed by reducing the training set. The experimental results show that our method is characterized by significantly higher speed than an ordinary SVM. Although typical use of SVM is still superior in terms of accuracy or other tested metrics, the difference is not significant. However, the proposed technique outperformed ordinary SVM when applied to Amazon customer reviews dataset with higher NPV and TPR. Execution time for the Stanford Twitter sentiment corpus dataset was 7.9–54x faster, and for Amazon customer reviews dataset – 9.8–15.35x faster than an ordinary SVM.

The presented method is a part of our hybrid sentiment recognition system which will be introduced in our future works. There are several directions to work on the proposed method. First we will focus on increasing the classification accuracy by selecting significant training data and by tuning parameters. Second, it will be integrated in the hybrid sentiment recognition system. Finally, the third direction for future work would be to experiment with a real-world data. We are also looking forward to thoroughly investigate other novel techniques for large-scale sentiment analysis.

References

1. Boser, B.E., Guyon, I.M., Vapnik, V.N.: A training algorithm for optimal margin classifiers. In: Proceedings of the Fifth Annual Workshop on Computational Learning Theory, pp. 144–152 (1992)
2. Chorowski, J., Wang, J., Zurada, J.M.: Review and performance comparison of SVM-and ELM-based classifiers. Neurocomputing **128**, 507–516 (2014)

3. Cortes, C., Vapnik, V.: Support-vector networks. Mach. Learn. **20**(3), 273–297 (1995)
4. Damaševičius, R.: Optimization of SVM parameters for recognition of regulatory DNA sequences. Top **18**(2), 339–353 (2010)
5. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: a library for large linear classification. J. Mach. Learn. Res. **9**(Aug), 1871–1874 (2008)
6. Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, 1, 12 (2009)
7. Guo, L., Boukir, S.: Fast data selection for SVM training using ensemble margin. Pattern Recogn. Lett. **51**, 112–119 (2015)
8. Graf, H.P., Cosatto, E., Bottou, L., Dourdanovic, I., Vapnik, V.: Parallel support vector machines: the cascade SVM. In: Advances in Neural Information Processing Systems, pp. 521–528 (2005)
9. Khairnar, J., Kinikar, M.: Machine learning algorithms for opinion mining and sentiment classification. Int. J. Sci. Res. Publ. **3**(6), 1–6 (2013)
10. Kharde, V., Sonawane, P.: Sentiment analysis of twitter data: a survey of techniques. arXiv preprint arXiv:1601.06971 (2016)
11. Korovkinas, K., Danėnas, P., Garšva, G.: SVM and Naïve Bayes classification ensemble method for sentiment analysis. Baltic J. Modern Comput. **5**(4), 398–409 (2017)
12. Le, B., Nguyen, H.: Twitter sentiment analysis using machine learning techniques. In: Le Thi, H.A., Nguyen, N.T., Do, T.V. (eds.) Advanced Computational Methods for Knowledge Engineering. AISC, vol. 358, pp. 279–289. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-17996-4_25
13. Lee, Y.J., Mangasarian, O.L.: RSVM: reduced support vector machines. In: Proceedings of the 2001 SIAM International Conference on Data Mining, pp. 1–17 (2001)
14. Lei, H., Govindaraju, V.: Speeding up multi-class SVM evaluation by PCA and feature selection. In: Feature Selection for Data Mining, 72 (2005)
15. Liu, M., Xu, C., Xu, C., Tao, D.: Fast SVM Trained by Divide-and-Conquer Anchors, pp. 2322–2328 (2017). https://doi.org/10.24963/ijcai.2017/323
16. Liu, J., Zio, E.: SVM hyperparameters tuning for recursive multi-step-ahead prediction. Neural Comput. Appl. **28**(12), 3749–3763 (2017)
17. Manek, A.S., Shenoy, P.D., Mohan, M.C., Venugopal, K.R.: Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM classifier. World Wide Web **20**(2), 135–154 (2017)
18. Mao, X., Fu, Z., Wu, O., Hu, W.: Fast kernel SVM training via support vector identification. In: 2016 23rd International Conference Pattern Recognition (ICPR), pp. 1554–1559 (2016)
19. Meyer, O., Bischl, B., Weihs, C.: Support vector machines on large data sets: simple parallel approaches. In: Spiliopoulou, M., Schmidt-Thieme, L., Janning, R. (eds.) Data Analysis, Machine Learning and Knowledge Discovery. SCDAKO, pp. 87–95. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-01595-8_10
20. Mourad, S., Tewfik, A., Vikalo, H.: Data subset selection for efficient SVM training. In: 2017 25th European Signal Processing Conference (EUSIPCO), pp. 833–837 (2017)
21. Nandan, M., Khargonekar, P.P., Talathi, S.S.: Fast SVM training using approximate extreme points. J. Mach. Learning Res. **15**(1), 59–98 (2014)
22. Osman, H., Ghafari, M., Nierstrasz, O.: Hyperparameter optimization to improve bug prediction accuracy. In: IEEE Workshop on Machine Learning Techniques for Software Quality Evaluation (MaLTeSQuE), pp. 33–38 (2017)

23. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL 2002 Conference on Empirical Methods in Natural Language Processing, vol. 10, pp. 79–86 (2002)
24. Pedregosa, F., et al.: Scikit-learn: machine learning in python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
25. Sammut, C., Webb, G.I. (eds.): Encyclopedia of Machine Learning. Springer, New York (2011)
26. Sunkad, Z.A.: Feature selection and hyperparameter optimization of SVM for human activity recognition. In: 2016 3rd International Conference Soft Computing & Machine Intelligence (ISCMI), pp. 104–109 (2016)
27. R Development Core Team: R: A language and environment for statistical computing, R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org
28. Wang, S., Li, Z., Liu, C., Zhang, X., Zhang, H.: Training data reduction to speed up SVM training. Appl. Intell. **41**(2), 405–420 (2014)

J48S: A Sequence Classification Approach to Text Analysis Based on Decision Trees

Andrea Brunello[1(✉)], Enrico Marzano[2], Angelo Montanari[1], and Guido Sciavicco[3]

[1] University of Udine, Via delle Scienze, 206, Udine, Italy
{andrea.brunello,angelo.montanari}@uniud.it
[2] Gap S.r.l.u., Via Tricesimo, 246, Udine, Italy
e.marzano@gapitalia.it
[3] University of Ferrara, Via Saragat, 1, Ferrara, Italy
guido.sciavicco@unife.it

Abstract. Sequences play a major role in the extraction of information from data. As an example, in business intelligence, they can be used to track the evolution of customer behaviors over time or to model relevant relationships. In this paper, we focus our attention on the domain of *contact centers*, where sequential data typically take the form of oral or written interactions, and word sequences often play a major role in text classification, and we investigate the connections between sequential data and text mining techniques. The main contribution of the paper is a new machine learning algorithm, called J48S, that associates semantic knowledge with telephone conversations. The proposed solution is based on the well-known C4.5 decision tree learner, and it is natively able to mix static, that is, numeric or categorical, data and sequential ones, such as texts, for classification purposes. The algorithm, evaluated in a real business setting, is shown to provide competitive classification performances compared with classical approaches, while generating highly interpretable models and effectively reducing the data preparation effort.

1 Introduction

Nowadays, most companies strive to extract relevant knowledge regarding their business. However, although such data may indeed be an important source of strategic information, they are sometimes stored in an unstructured form, as, for instance, in the case of text or audio flows. Specifically, the ability to analyze conversational data plays a major role in contact centers, where the core part of the business still focuses on the management of oral interactions [1]. Among the several benefits that *speech analytics* can deliver in such a domain, we mention their possible use in tracking down problematic calls [2,3], and, more generally, in the development of an end-to-end solution to conversation analysis [4].

Since call conversation transcripts may be regarded as (a kind of) sequential data, and word sequences typically play a major role in text classification tasks, in this paper we propose a novel decision tree induction algorithm based on the

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 240–256, 2018.
https://doi.org/10.1007/978-3-319-99972-2_19

well-known inducer C4.5 [5] and the algorithm VGEN for the extraction of frequent patterns [6]. The proposed algorithm is able to mix the use of categorical, numeric, and sequential attributes during the same execution cycle, leading to several benefits, most importantly: *(i)* a simplification of the data preparation phase, especially in the preprocessing of textual attributes, and *(ii)* a very high interpretability of the generated models.

The proposed solution has been tested on recorded agent-side call conversations produced by a wide-range survey campaign run on Italian subjects by Gap S.r.l.u., an Italian business process outsourcer specialized in contact center services. The considered task consists in detecting relevant contents in call conversation transcripts, by determining the presence or absence of a predefined set of semantic tags. The ability of tagging telephone conversations constitutes a fundamental step towards the overall company's goal of introducing a reliable speech analytics solution with a real operational impact on company processes, such as the assessment of the training level of human employees or the assignment of a quality score to the calls. The results achieved by the decision trees are shown to be competitive with respect to the ones obtained from more traditional approaches, relying on *ngrams*, but with the added benefit of providing a much more interpretable model. Moreover, the developed algorithm is not limited to text classification tasks, but it configures itself as a more general solution, that can be applied to any kind of classification problem containing sequential data as well as static attributes.

The main focus of the work is proposing J48S, along with its implementation details. The paper is organized as follows: Sect. 2 presents the call center domain and introduces the analysis task on which the novel decision tree is tested, i.e., the tagging of telephone conversation transcripts. Section 3 describes the main algorithms on which J48S is based. Section 4 thoroughly describes the technical aspects of J48S. Finally, Sect. 5 presents the application of the algorithm to a real test case, before concluding.

2 Domain and Problem Description

Telephone call centers, as an integral part of many businesses, are an increasingly important part of today's business world, acting as a primary customer-facing channel for firms in many different industries, and employing millions of *operators* (also called *agents*) across the globe [7]. Call centers can be categorized according to different parameters. They can be classified with respect to the functions they provide, e.g., customer services, telemarketing, emergency response, help-desk, and order taking. More importantly, they can be partitioned into *inbound* (answering to incoming calls) and *outbound* (making calls, - e.g., to ask a set of predefined questions in a telephone survey). Regardless the type, as it operates, a large call center generates vast amounts of data, both structured (as in the case of the tracking of called numbers, call durations, participating agents, etc.) and unstructured (consider, for instance, textual notes written by the agents, call recordings, etc.).

In this paper, we focus on the analysis of agent-side calls originated in the context of an outbound survey service, carried out by Gap. Agents performing outbound calls typically ought to complete a predefined script in order for a call to be considered successful, and such a rigid structure makes it easier to analyze a conversation and to determine if the agent has followed all the required steps or not. In order to do so, machine learning may be employed to learn models capable of determining the presence or absence of certain *tags* in call transcripts. Such tags have a precise meaning in the domain, and track the presence of specific contents in the conversations. Although such an analysis may seem trivial, it has very deep practical implications for the company. As a matter of fact, typically, the work of an agent is manually checked by supervising staff by simply listening to a random number of calls made by him/her; clearly, such an operation is time-consuming and it requires a lot of enterprise resources. An automatic analysis module may greatly simplify it by identifying "problematic" calls that require further attention, thus reducing the time needed for verification, and increasing the overall efficacy of the process. Based on the analysis tasks, the supervising staff may then identify agents who are in need of further training, and also the specific deficiencies that have to be solved. Overall, the entire process constitutes a building block of a larger analytics framework in use by the company.

3 Background

Our algorithm is inspired by [8], where the authors propose an ad-hoc decision tree induction algorithm for classifying purely sequential data. However, we rely on a well-know and performing decision tree inducing algorithm, such as J48, which is Weka's implementation [9] of the algorithm known as C4.5 [5], and we develop a technique to extend it in order to support and exploit sequential data. As we shall see, our solution makes a combined use of categorical, numeric, and sequential attributes during the same execution cycle.

3.1 Decision Trees

As it is commonly recognized, to-date decision trees are of primary importance among classification models [9]. They owe their popularity mainly to the facts that they can be trained and applied efficiently even on big datasets, and they are easily interpretable, meaning that they are not only useful for prediction per se, but also for helping domain experts to discover new insights into the data. The latter point is of extreme importance in domains in which understanding the classification process is as important as the accuracy of the classification itself, such in the case of production business systems. Consider a dataset of training instances, each characterized by the values it takes on a set of features, and labeled by a class value. A typical decision tree is then constructed in a recursive manner, following the traditional *Top Down Induction of Decision Trees* (TDIDT) approach: starting from the root, at each node the attribute that best partitions the training data, according to a predefined score, is chosen as a test to guide the partitioning of instances into child nodes. The process continues until a sufficiently high degree of purity (with respect to the target class), or a minimum cardinality constraint (with respect to the number of instances reaching the node), is achieved in the generated partitions.

In this paper, we make use of the well-known and widely-adopted J48/C4.5 decision tree learning algorithm, which adopts the *information gain* (an entropy-based criterion) as the scoring strategy to guide the splitting process [5]. Given a set of observable values $V = (v_1, v_2, \ldots, v_n)$, with associated probabilities $P = (p_1, p_2, \ldots, p_n)$, the *information conveyed by* P, or the *entropy of* P, is defined as

$$E(P) = -\sum_{i=1}^{n} p_i * log_2(p_i). \tag{1}$$

If a dataset T of instances is partitioned into disjoint exhaustive subsets $C_1, \ldots,$ C_k on the basis of the class value, then the information needed to identify the class of an element of T is

$$Info(T) = E(P), \text{ where } P = (|C_1|/|T|, |C_2|/|T|, \ldots, |C_k|/|T|).$$

Intuitively, the purer T is with respect to the class values, the lower the entropy, and the easier to identify the class of an instance. Given T and a categorical attribute X of the dataset (non class), we may partition the instances into subsets $T_1, T_2, \ldots, T_j$ on the basis of the value they take on X. Then, the information needed to identify the class of an element of T becomes the weighted average of the information needed to identify the class of an element of T_i for each $i = 1, \ldots, j$, that is,

$$Info(X, T) = \sum_{i=1}^{j} (|T_i|/|T|) * Info(T_i). \tag{2}$$

We may now define *information gain* as the difference between the information needed to identify an element of T and the information needed to identify such an element after the value it takes on X has been obtained:

$$Gain(X, T) = Info(T) - Info(X, T). \tag{3}$$

Thus, the information gain represents precisely the gain in information due to attribute X. Starting from the root, at each node of the tree C4.5 chooses to branch on the attribute of the data which it considers best, namely the one with the highest *information gain ratio*, defined as

$$GainRatio(X, T) = \frac{Gain(X, T)}{SplitInfo(X, T)}, \tag{4}$$

where

$$SplitInfo(X, T) = E(|T_1|/|T|, \ldots, |T_j|/|T|). \tag{5}$$

is the *split information* metric, that takes into account the number and size of branches that would be generated if the split on attribute X is chosen. Introducing $SplitInfo$ corrects the bias that the information gain has towards highly branching attributes (as an example, consider a dataset of people containing the attribute *social security number*, having unique values; a split on that attribute would produce single-instance subsets, leading to the highest possible *information gain*).

C4.5 also supports numeric attributes. In order to branch on a numeric attribute, the algorithm identifies a threshold and then splits the instances into those whose attribute value is *above* the threshold and those that are *less than or equal to* it [10]. This means that numeric splits are always binary, contrary to the categorical ones, which generate a number of children equal to the number of distinct values. Let us now consider the selection of a threshold for a numeric attribute A, belonging to a dataset D. If there are N distinct values of A in D, then there are $(N-1)$ thresholds that could be used *differently* for a binary test on A during the tree construction phase, since any threshold between two values will have the same effect in dividing the instances. Each threshold gives unique subsets D_1 and D_2, so in this case the value of the gain ratio is not only a function of the attribute, but also of the threshold. Thus, a straightforward extension to the categorical approach is to evaluate the ratio for each of the $(N-1)$ split points, choosing the one that leads to the highest value. The gain ratio value for that particular split point is considered to be the gain ratio value for the attribute. To overcome the overfitting problems of this approach, an *MDL-based adjustment* to the information gain for splits on numeric attributes was included in [10]: if there are K candidate splits on a certain numeric attribute at the node currently considered for splitting, $log_2(K)/M$ is subtracted from the information gain, where M is the number of instances at the node.

3.2 Sequential Pattern Mining

Sequential pattern mining is a popular data mining task aimed at discovering "interesting" patterns in sequences [11,12], that is, patterns that can prove to be useful for supervised or unsupervised learning tasks. In particular, we are interested in finding *frequent patterns*, which are concatenations of symbols that frequently appear in a set of sequences, with a frequency that is no less than a user-specified *minimum support threshold* [13]. Several sequential pattern mining algorithms have been proposed over the years, such as *PrefixSpan* [14], *SPADE* [15], and *SPAM* [16]. However, a drawback of such algorithms is that they typically generate a large amount of patterns, which may be redundant. This is problematic for two reasons: *(i)* it makes it difficult for a user to gain any insight based on the generated output, and also to exploit the patterns for further data mining tasks, and *(ii)* too large sets of patterns may imply too much resources consumed and too high computation time. In order to reduce the computational burden of the mining tasks, and also to present a reduced set of patterns to the user, some concise representations of frequent sequential patterns have been designed. *A frequent closed sequential pattern* is a frequent sequential pattern such that it is not included in any other sequential pattern having exactly the same frequency, that is, it is "maximal". Algorithms that have been developed to extract such patterns include *CloSpan* [17], *BIDE* [18], *ClaSP* [19], and *CM-ClaSP* [20]. Another possible solution is given by *sequential generator patterns*. Opposite to the concept of closed patterns, generators are "minimal", meaning that a pattern is considered to be a generator if it does not exist any pattern which is contained in it and has the same frequency. According to the MDL principle [21], generators are preferable over closed patterns for model selection and classification, since they may be less prone to overfitting effects [22]. There also exists a very recent solution capable of extracting patterns according to different concise representations during the same execution cycle [23].

In this work, we use the algorithm VGEN [6], which, at the best of our knowledge, represents the state-of-the-art in the extraction of general sequential generator patterns. VGEN operates in a top-down fashion. A pattern is considered as a sequence of itemsets, which are in turn unordered sets of distinct items. An item is an element that may appear in a sequence belonging the considered dataset. Intuitively, items that belong to the same itemset are considered to occur at the same time. The algorithm starts by listing all single-itemset, single-item frequent patterns, and then it grows them by means of the so-called *s-extension* operations (in which a new itemset is enqueued to the pattern) and *i-extension* operations (in which a new item is added to the last itemset), ensuring also that only generator patterns are produced. Of course, as a pattern grows, its support (number of instances in the dataset that contain the pattern) decreases, and the growing phase continues until a minimum support threshold is reached, which may be specified by the user. The algorithm is also capable of extracting non strictly contiguous patterns, by specifying a maximum gap tolerance between the itemsets.

4 J48S Algorithm Description

In this section we present a novel decision tree learning algorithm based on J48 (Weka's implementation of C4.5, release 8 [9]), and the VGEN pattern extraction algorithm. All the modifications that have been made in order to integrate the two algorithms together are thoroughly described.

4.1 VGEN

As we have recalled, VGEN produces frequent generator patterns in a top-down fashion, until a user-defined minimum support threshold is reached. Since we are not interested in obtaining frequent patterns, but rather discriminative patterns that may help in the classification process, we integrated the *information gain* criterion into the algorithm. For the sake of simplicity, let us consider the two-class classification problem; the concepts can be easily extended to multi-class classification by means of an one-vs-all approach.

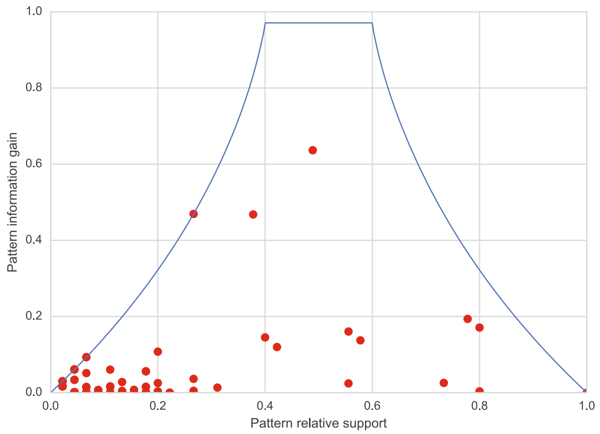


Fig. 1. Pattern information gain and its theoretical upper bound with respect to the support on a randomly generated dataset (Color figure online)

First, note that the information gain of a pattern may be calculated by partitioning the instances in the dataset into those that satisfy it, and those which do not. As observed in [24], the information gain can be considered as a function of the support of the pattern: patterns having a very high or very low support are typically not very informative, the former being simply too common, and the latter having a too limited coverage in the dataset. On the basis of these assumptions, in [24], a strategy to derive an upper bound to the information gain based on the support of the pattern is presented. The behavior of such an upper bound is illustrated as the solid curve in Fig. 1, which considers data taken from a randomly generated binary class dataset: as expected, from left to right, the upper bound increases as long as the support increases, until it peaks, and then starts decreasing as the relative support reaches 1. The graph also shows the information gain and support of each pattern extracted by VGEN on the same dataset (red dots). Thus, given the upper bound calculation strategy, we are able of giving a stopping criterion for the patter growing process. During the pattern generation process we store in a dedicated variable the best information gain found so far. Then, given a pattern, we decide if it is worth growing as follows: if the support of the pattern corresponds to a point in the curve where the derivative is positive or zero, then growing it may only decrease its upper bound or leave it unchanged. Thus, if the best information gain found so far is already greater or equal to the information gain upper bound of the pattern, we may stop its growing process. This strategy constitutes a second stopping condition, which is employed together with the minimum support threshold method.

VGEN has also been modified with respect to its output. As we shall see, VGEN is called at each node during the tree construction phase to obtain a candidate pattern to guide the split operation. So, we are not interested in obtaining a set of patterns (as returned by the standard VGEN algorithm), but rather in the single most useful pattern at that stage. Note, however, that sometimes we cannot just take the most informative one, as it might be too complex and thus overfit training data. Hence, we proceed as follows. We store the best pattern (in terms of information gain) for each encountered pattern size. Then, at the end of the algorithm, the following formula (again, inspired by the MDL principle [21]), is evaluated for each pattern:

$$W * (1 - pattern_{rIG}) + (1 - W) * pattern_{rlen} \tag{6}$$

where $W \in [0, 1]$ is a weight that can be customized by the user, $pattern_{rIG} \in [0, 1]$ is the relative information gain of the pattern with respect to the highest information gain observed, and $pattern_{rlen} \in [0, 1]$ is the relative length of the pattern, with respect to the longest pattern that has been discovered (in number of items). We then select the pattern the minimizes the value of such formula, thus, intuitively, the larger the user sets the weight, the longer and more accurate (on the training set) the extracted pattern will be. On the contrary, selecting a low weight should lead to a shorter, and more general pattern being selected. In essence, setting W allows us to control the level of abstraction of the extracted patterns.

4.2 J48

The decision tree J48 is capable of handling both categorical and numeric attributes. Our extension J48S adds the possibility of managing also sequences, which are represented as properly formatted strings. Our design is inspired by [8], in which the authors propose a decision tree construction strategy for sequential data. Our work improves [8] mainly over three aspects: *(i)* we rely on a well-established algorithm such as J48, instead of designing a new one; *(ii)* our approach is capable of mixing the usage of sequential and classical (i.e., categorical or numeric) data on the same execution cycle; *(iii)* our implementation allows the tuning of the abstraction level of the extracted patterns.

Our first modification concerns the splitting criterion. Instead of relying on the gain ratio, we use the so-called *normalized gain* [25], which is defined as:

$$NormIG(X, T) = \frac{Gain(X, T)}{log_2 n},\ n \geq 2 \tag{7}$$

where n is the split arity. Note that this allows for a seamless integration between the decision tree learner and the modified VGEN algorithm, since the normalized gain is exactly equal to the information gain for binary splits, as in the case of a pattern-based split (recall that the criterion is the presence or absence of the pattern in the instance). At each node of the tree, the learner determines the most informative attribute among the categorical and numerical ones. Then, it calls VGEN along with two fundamental parameters: *(i)* the user-defined weight for controlling the abstraction level of the extracted pattern, W; and *(ii)* the normalized gain of the best attribute found so far, so that it can be used to prune the pattern search space as discussed in Sect. 4.1. Once VGEN exits, if

Table 1. Custom parameters in J48S

Parameter name	Default	Description
maxGap	2	max gap allowance between itemsets (1 = contiguous)
maxPatternLength	20	max length of a pattern, in number of items
maxTime	30	max allowed running time of the algorithm, per call
minSupport	0.5	min support of a pattern
patternWeight	0.5	weight used in VGEN for the extraction of the result
useIGPruning	True	use information gain pruning of the pattern search space

a pattern is returned, its normalized gain is compared with the best previously found. If it is better, then, a binary split is created on the basis of the presence or absence of the pattern. Otherwise, the best numeric or categorical attribute is used.

Table 1 lists the six parameters that have been added to J48S, with respect to the original J48. Observe, in particular, the role of *maxGap*: it allows to tolerate gaps between itemsets which, in a text classification context, means that it is possible to extract and apply ngrams capable of tolerating the presence of some noisy or irrelevant words. Note that when *maxGap* is set to values larger than 1, it greatly affects the computation time of the pattern extraction phase: in general the higher its value, the slower the algorithm. As an example, Fig. 2 presents an archetypal J48S decision tree, built by integrating the extension to Weka data mining suite [9]. The reference dataset is characterized by three features: *sequence_attribute*, which is of type string and represents a sequence of itemsets; *attribute_numeric* which has an integer value; *attribute_nominal*, which may only take one out of a predefined set of values. The class is binary, and has two labels: *class_0* and *class_1*. The first test made by the tree is whether the given sequence contains the pattern $(A, B) > D$ or not, meaning that there should be an itemset containing A, B, followed (within the maximum gap constraint) by an itemset containing D. If this is the case, the instance is labeled with *class_0*, otherwise the tree proceeds by testing on the numeric and nominal attributes.

```
sequence_attribute !contains (A,B)>D
|  attribute_numeric <= 20: class_1
|  attribute_ numeric > 20
|  |  attribute_nominal = value_1: class_0
|  |  attribute_nominal = value_2: class_1
sequence_ attribute contains (A,B)>D: class_0
```

Fig. 2. A typical J48S decision tree built by means of Weka

5 Experiments: Tagging Call Transcripts

In this section we presents the setup and the results of the experimentation carried out with J48S, with the aim of tagging call transcripts with relevant tags. For the considered outbound survey service, the defined tags are:

- *age:* the agent asked the interviewed person his/her age;
- *call_permission:* the agent asked the called person for the permission to conduct the survey;
- *duration_info:* the agent informed the called person about the duration of the survey;
- *family_unit:* the agent asked the called person about his/her family unit;
- *greeting_initial:* the agent introduced himself/herself correctly at the beginning of the phone call;
- *greeting_final:* the agent pronounced the scripted goodbye phrases;
- *person_identity:* the agent asked the called person for a confirmation of his/her identity;
- *privacy:* the agent informed the called person about the privacy implications of the phone call;
- *profession:* the agent asked the interviewed person about his/her job;
- *question_1:* the agent asked the first question of the survey;
- *question_2:* the agent asked the second question of the survey;
- *question_3:* the agent asked the third question of the survey.

In Table 2 we present examples of real transcriptions with the associated tags. It should be observed that tags are general concepts, which are independent from the specific service under consideration. For example, the *privacy* tag may apply to different services.

Table 2. Real transcriptions with the associated tags. Data has been anonymized, and punctuation has been added for ease of reading

Phrase (Italian)	Phrase (English)	Tags
Si pronto buongiorno sono X dalla X di X, parlo con la signora X?	Hello, my name is X and I am calling from X of X, am I talking with Mrs X?	greeting_initial, person_identity
Lei è pensionato. Ultima domanda, senta, a livello statistico la data solo di nascita... millenovecento...?	You are retired. Last question, listen, statistically, the birth date only... nineteen hundred...?	age, profession
Ho capito. Posso chiederle il nome di battesimo?	Understood. May I ask you for your first name?	person_identity
Mi permette? Trenta secondi, tre domande velocissime...	May I? Thirty seconds, three quick questions...	duration_info

Table 3. Number and percentage of positive and negative tags

Tag name	#positive	%positive	#negative	%negative
age	638	13.1	4246	86.9
call_permission	565	11.6	4319	88.4
duration_info	491	10.1	4393	89.9
family_unit	506	10.4	4378	89.6
greeting_initial	560	11.5	4324	88.5
greeting_final	453	9.3	4431	90.7
person_identity	600	12.3	4284	87.7
privacy	440	9.0	4444	91.0
profession	391	8.0	4493	92.0
question_1	516	10.6	4368	89.4
question_2	496	10.2	4388	89.8
question_3	500	10.2	4384	89.8

To allow for the training and evaluation of the machine learning models, a set of 4884 text chunks (phrases delimited by silence pauses) transcribed from 482 distinct outbound calls by means of an internally-developed solution based on the Kaldi toolkit [26], have been manually tagged by domain experts, so that each instance is characterized by the transcription, and by a list of boolean attributes that track the presence or absence of each specific tag. In Table 3 we report the number of positive and negative instances for each tag. The resulting dataset has been split into a training (75%, 3696 instances) and a test (25%, 1188 instances) set, according to a *stratified random sampling by group* approach, where each single session is a group on its own. This allowed us not to fragment chunks belonging to a single session between the two sets and, moreover, to preserve the tag distribution between them. Finally, observe that the reported word error rate (WER) for the Kaldi model is 28.77% (for comparison, consider that a model based on the Google Speech API has been capable of obtaining a WER of 18.70% on the same dataset). Although such an error rate may seem high, it is actually low enough to make the analysis tasks needed by the company possible[1].

5.1 Data Preparation

The training and test datasets have been prepared as follows, starting from the data described in Sect. 5. First of all, the text has been converted to lowercase and all non-alphabetical characters have been removed from each chunk. Then, all Italian stopwords have been removed. Subsequently, the stemming algorithm *PORTER2* for Italian language has been applied. All operations have been carried out by means of custom made *Python* scripts, using the NLTK library.

[1] A detailed account of these aspects is the object of a forthcoming work about the whole speech analytics process.

A training and a test dataset have been created for each tag (for a total of 12 dataset couples), where each of the instances is characterized by just two attributes: the processed text of the chunk, and an attribute identifying the presence or absence of the specific tag (class).

5.2 Training of the Model

J48S has been implemented within the Weka data mining suite [9]. For the training phase, all parameters have been left with their default values, except for: *(i) minSupport*, which has been set to 0.01 (1%), in order to allow for the extraction of uncommon patterns, and *(ii) patternWeight* which has been set to 1, considering that the survey service requires very specific phrases to be pronounced by the agents.

5.3 Evaluation of Tagging Performances

Each of the 1188 chunks in the test set has been evaluated against the presence of every possible tag, looking for the concordance between its manual annotations and the ones given by the corresponding decision tree. The following standard metrics have been used to evaluate the performances:

- *accuracy*, that is, the fraction of times when a tag has been correctly identified in a chunk as present or absent;
- *precision*, which is the fraction of chunks in which a specific tag has been identified as present by the method, and in which the tag is indeed present;
- *recall*, that reports the proportion of chunks presenting the specific tag, that have been in fact identified as such;
- *true negative rate* (TNR), which shows the proportion of chunks not presenting the specific tag, that have been classified as negative by the method;

As the results presented in Table 4 show, the proposed approach is capable of reaching a very high accuracy in tagging chunks. Such a result alone is however not indicative of the true performance, since it is biased by the large disproportion between positive (present) and negative (not present) instances of the tags. More useful indicators are represented by the true negative rate, precision and, especially, recall, which reveals the tags that are most difficult to identify, and is of most importance given the specific business task at hand. Anyway, highly satisfactory results are obtained for most of the tags, except for *call_permission*, which falls slightly under 0.65 precision and recall. For example, consider the tag *greeting_final*. A typical, well-recited, ending phrase for the service should be of the form: *"Ok, we have finished. Thank you for the cooperation and good day, also on behalf of [company_name]. If you allow me, I inform you that [company_name] may recontact you"*. Figure 3 presents the model that recognizes the presence of the tag (words have been unstemmed for the ease of reading). As can be seen, besides the chosen representation, the tree is very simple and intuitive, having just 9 leaves. For the sake of explanation, the tree already discards conversations which do not contain the words *recontact* and *inform* and

Table 4. Tagging performances of J48S

Tag name	Accuracy	Precision	Recall	TNR
age	0.9655	0.9241	0.8171	0.9893
call_permission	0.9167	0.6500	0.6454	0.9532
duration_info	0.9739	0.9008	0.8516	0.9887
family_unit	0.9722	0.8779	0.8712	0.9848
greeting_initial	0.9739	0.9397	0.8195	0.9934
greeting_final	0.9916	0.9823	0.9328	0.9981
person_identity	0.9428	0.7851	0.6934	0.9753
privacy	0.9916	0.9550	0.9550	0.9554
profession	0.9840	0.8969	0.9063	0.9908
question_1	0.9857	0.9688	0.9051	0.9962
question_2	0.9815	0.9153	0.9000	0.9906
question_3	0.9798	0.9237	0.8790	0.9915
Macro average	**0.9716**	**0.8933**	**0.8480**	**0.9839**

```
transcription !contains recontact
|   transcription !contains thank>cooperation
|   |   transcription !contains inform: 0
|   |   transcription contains inform
|   |   |   transcription !contains inform>company_name: 0
|   |   |   transcription contains inform>company_name
|   |   |   |   transcription !contains allow: 0
|   |   |   |   transcription contains allow: 1
|   transcription contains thank>cooperation
|   |   transcription !contains ok>finish>thank>behalf
|   |   |   transcription !contains allow>inform
|   |   |   |   transcription !contains cooperation>day: 0
|   |   |   |   transcription contains cooperation >day: 1
|   |   |   transcription contains allow>inform: 1
|   |   transcription contains ok>finish>thank>behalf: 1
transcription contains recontact: 1!
```

Fig. 3. J48S decision tree recognizing the presence of the tag *greeting_final*

the sequence *thank>cooperation*, which are in fact very important steps of the scripted phrase, that should not be neglected. As for the complexity of the training phase, the considered tree has been built in less than 30 s on an i5 powered laptop, while its application to a new instance is almost instantaneous.

For the sake of comparison, also logistic regression has been used to train models capable of recognizing tags in the chunks. Although such a choice may seem somehow different from what is typically done in the literature, where support vector machines are often employed for text classification purposes, consider that interpretability of the generated models is a fundamental requirement for the company. Three training and test dataset pairs have then been created for each keyword, considering different ngram lengths and again a frequency threshold of 1%: unigrams (leading to a total of 177 attributes, plus the class), bigrams (154 attributes, plus the class) and trigrams (102 attributes, plus the class). Given the 12 possible tags, and the 3 different ngram lengths that have been considered, a total of 36 training and test dataset couples have been generated. Logistic models have been trained by means of Weka's *logistic* algorithm, employing also a feature selection step based on the multivariate, classifier-independent, filter *CfsSubsetEval* [27], which reduced the number of attributes to an average of 12. Surprisingly, the experimental results suggested that the unigram-based models are the most effective in tagging the transcripts. This may be explained in two ways: *(i)* there might not be a sufficient number of instances in the datasets to justify the adoption of bigrams and trigrams, or *(ii)* the presence or absence of a tag in calls generated in the context of the specific service that has been considered is not a difficult concept to learn, so unigrams are sufficient for the task. Table 5 reports the detailed results for the unigram models.

Table 5. Tagging performance with logistic regression

Keyword name	Accuracy	Precision	Recall	TNR
age	0.9663	0.8974	0.8537	0.9844
call_permission	0.9310	0.7980	0.5603	0.9809
duration_info	0.9672	0.9083	0.7734	0.9906
family_unit	0.9630	0.8929	0.7576	0.9886
greeting_initial	0.9773	0.9206	0.8722	0.9905
greeting_final	0.9941	0.9912	0.9496	0.9991
person_identity	0.9369	0.7500	0.6788	0.9705
privacy	0.9907	0.9545	0.9459	0.9954
profession	0.9874	0.8932	0.9583	0.9899
question_1	0.9840	0.9683	0.8905	0.9962
question_2	0.9840	0.9244	0.9167	0.9916
question_3	0.9848	0.9417	0.9113	0.9934
Macro average	**0.9722**	**0.9033**	**0.8390**	**0.9893**

Considering the (macro) average values of the logistic approach, most of the performance metrics are very close to the ones provided by J48S, with the exception of recall (once again, the most important metric in this context), which is lower than the one exhibited by the decision tree, somehow balanced by a slightly higher precision. Thus, J48S is capable of providing results comparable to the ones of the best approach based on logistic regression. Moreover, it is worth mentioning that J48S's results have been obtained with default parameters (except for the minimum support value and pattern weight, which have been set following domain expertise), thus, even better performances may be obtained through a dedicated tuning phase. But, most importantly, data preparation for J48S is considerably simpler than for the ngrams-based regression, since the decision tree is capable of directly exploiting textual information, allowing one to avoid the ngram preprocessing step, and the related a-priori choices of the ngram sizes and frequencies altogether (observe also that ready-to-process datasets for J48S had precisely two attributes, while those for the ngrams-based approach varied from 103 to 178 attributes). In addition, sequential patterns may be more resistant to noise in the data than bigrams and trigrams, as they are capable of skipping irrelevant words thanks to the *maxGap* parameter. Finally, and most importantly, the constructed trees are much more comprehensible to (non computer scientists) domain experts than regression models.

6 Conclusions

In this paper, J48S, a novel decision tree learning algorithm capable of mixing the use of categorical, numeric and sequential data has been presented. One of its possible applications is the classification of textual data and, in this work, it has been applied to the problem of tagging call conversation transcripts in a real business setting. The experiments show that the proposed algorithm is capable of matching the results provided by more traditional approaches, but with several added benefits, most importantly: a reduction in the complexity of the preprocessing phase of textual data, and the generation of highly interpretable models.

As for future work, an extension to the algorithm J48S with the capability of managing continuous, numerical time series is under study. This would open a large amount of new learning possibilities and, among them, classification of telephone conversations is still a major application, since such task presents all kinds of data that would be supported by the extended algorithm: static attributes (e.g., the length of the transcription, the average conversation volume, etc.), sequential attributes (such as the conversation transcript), and time series (e.g., tracking the evolution in time of the sound levels).

References

1. Saberi, M., Khadeer Hussain, O., Chang, E.: Past, present and future of contact centers: a literature review. Bus. Process. Manag. J. **23**(3), 574–597 (2017)
2. Cailliau, F., Cavet, A.: Mining automatic speech transcripts for the retrieval of problematic calls. In: Gelbukh, A. (ed.) CICLing 2013. LNCS, vol. 7817, pp. 83–95. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-37256-8_8
3. Pandharipande, M.A., Kopparapu, S.K.: A novel approach to identify problematic call center conversations. In: Ninth International Joint Conference on Computer Science and Software Engineering (JCSSE 2012), pp. 1–5 (2012)
4. Garnier-Rizet, M., et al.: CallSurf: automatic transcription, indexing and structuration of call center conversational speech for knowledge extraction and query by content. In: Sixth International Conference on Language Resources and Evaluation (LREC 2008), pp. 2623–2628 (2008)

5. Quinlan, J.R.: Simplifying decision trees. Int. J. Man Mach. Stud. **27**(3), 221–234 (1987)
6. Fournier-Viger, P., Gomariz, A., Šebek, M., Hlosta, M.: VGEN: fast vertical mining of sequential generator patterns. In: Bellatreche, L., Mohania, M.K. (eds.) DaWaK 2014. LNCS, vol. 8646, pp. 476–488. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10160-6_42
7. Gans, N., Koole, G., Mandelbaum, A.: Telephone call centers: tutorial, review, and research prospects. Manuf. Serv. Oper. Manag. **5**(2), 79–141 (2003)
8. Fan, W., et al.: Direct mining of discriminative and essential frequent patterns via model-based search tree. In: Fourteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2008), pp. 230–238 (2008)
9. Witten, I.H., Frank, E., Hall, M.A., Pal, C.J.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, San Francisco (2016)
10. Quinlan, J.R.: Improved use of continuous attributes in C4.5. J. Artif. Intell. Res. **4**(1), 77–90 (1996)
11. Mabroukeh, N.R., Ezeife, C.I.: A taxonomy of sequential pattern mining algorithms. ACM Comput. Surv. **43**(1), 1–41 (2010)
12. Fournier-Viger, P., Lin, J.C.W., Kiran, R.U., Koh, Y.S., Thomas, R.: A survey of sequential pattern mining. Data Sci. Pattern Recognit. **1**(1), 54–77 (2017)
13. Agrawal, R., Srikant, R.: Mining sequential patterns. In: Eleventh IEEE International Conference on Data Engineering (ICDE 1995), pp. 3–14 (1995)
14. Pei, J., et al.: Mining sequential patterns by pattern-growth: the prefixspan approach. IEEE Trans. Knowl. Data Eng. **16**(11), 1424–1440 (2004)
15. Zaki, M.J.: Spade: an efficient algorithm for mining frequent sequences. Mach. Learn. **42**(1), 31–60 (2001)
16. Ayres, J., Flannick, J., Gehrke, J., Yiu, T.: Sequential pattern mining using a bitmap representation. In: Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2002), pp. 429–435 (2002)
17. Yan, X., Han, J., Afshar, R.: CloSpan: mining closed sequential patterns in large datasets. In: 2003 SIAM International Conference on Data Mining (SIAM 2003), pp. 166–177 (2003)
18. Wang, J., Han, J.: BIDE: efficient mining of frequent closed sequences. In: Twentieth IEEE International Conference on Data Engineering (ICDE 2004), pp. 79–90 (2004)
19. Gomariz, A., Campos, M., Marin, R., Goethals, B.: ClaSP: an efficient algorithm for mining frequent closed sequences. In: Pei, J., Tseng, V.S., Cao, L., Motoda, H., Xu, G. (eds.) PAKDD 2013. LNCS, vol. 7818, pp. 50–61. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-37453-1_5
20. Fournier-Viger, P., Gomariz, A., Campos, M., Thomas, R.: Fast vertical mining of sequential patterns using co-occurrence information. In: Tseng, V.S., Ho, T.B., Zhou, Z.-H., Chen, A.L.P., Kao, H.-Y. (eds.) PAKDD 2014. LNCS, vol. 8443, pp. 40–52. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-06608-0_4
21. Rissanen, J.: Modeling by shortest data description. Automatica **14**(5), 465–471 (1978)
22. Lo, D., Khoo, S.C., Li, J.: Mining and ranking generators of sequential patterns. In: 2008 SIAM International Conference on Data Mining (SIAM 2008), pp. 553–564 (2008)
23. Duong, H., Truong, T., Le, B.: Efficient algorithms for simultaneously mining concise representations of sequential patterns based on extended pruning conditions. Eng. Appl. Artif. Intell. **67**, 197–210 (2018)

24. Cheng, H., Yan, X., Han, J., Hsu, C.W.: Discriminative frequent pattern analysis for effective classification. In: Twenty-Third IEEE International Conference on Data Engineering (ICDE 2007), pp. 716–725 (2007)
25. Jun, B.H., Kim, C.S., Song, H.Y., Kim, J.: A new criterion in selection and discretization of attributes for the generation of decision trees. IEEE Trans. Pattern Anal. Mach. Intell. **19**(12), 1371–1375 (1997)
26. Povey, D., et al.: The Kaldi speech recognition toolkit. In: IEEE 2011 Workshop on Automatic Speech Recognition and Understanding (ASRU 2011), pp. 1–4 (2011)
27. Hall, M.A.: Correlation-based feature selection for machine learning. Ph.D. thesis, The University of Waikato (1999)

TabbyPDF: Web-Based System for PDF Table Extraction

Alexey Shigarov[1,2(✉)], Andrey Altaev[1], Andrey Mikhailov[1], Viacheslav Paramonov[1,2], and Evgeniy Cherkashin[1,2]

[1] Matrosov Institute for System Dynamics and Control Theory of SB RAS, 134 Lermontov st., Irkutsk, Russia
shigarov@icc.ru
http://cells.icc.ru
[2] Institute of Mathematics, Economics and Informatics, Irkutsk State University, 20 Gagarin blvd., Irkutsk, Russia

Abstract. PDF is one of the most widespread ways to represent non-editable documents. Many of PDF documents are machine-readable but remain untagged. They have no tags for identifying layout items such as paragraphs, columns, or tables. One of the important challenges with these documents is how to extract tabular data from them. The paper presents a novel web-based system for extracting tables located in untagged PDF documents with a complex layout, for recovering their cell structures, and for exporting them into a tagged form (e.g. in CSV or HTML format). The system uses a heuristic-based approach to table detection and structure recognition. It mainly relies on recovering a human reading order of text, including document paragraphs and table cells. A prototype of the system was evaluated, using the methodology and dataset of "ICDAR 2013 Table Competition". The standard metric F-score is 93.64% for the structure recognition phase and 83.18% for the table extraction with automatic table detection. The results are comparable with the state-of-the-art academic solutions.

Keywords: Table understanding · Table extraction · Table detection
Table recognition · Document analysis · PDF accessibility

1 Introduction

PDF is one of the most widespread ways to represent non-editable documents. P. Ydens guesstimates in his keynote address[1] on PDF Technical Conference 2015 that there might be 2.5 trillion PDF files created each year. Many of PDF documents are machine-readable but remain untagged. They have no tags for identifying layout items, including running titles, sections, paragraphs, figures, lists, and tables. A recent estimation reported by Nganji [12] claims that 95.5% of all scientific articles published by four leading publishers are untagged PDF

[1] https://www.pdfa.org/pdf-in-2016-broader-deeper-richer.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 257–269, 2018.
https://doi.org/10.1007/978-3-319-99972-2_20

documents. One of the important challenges with these documents is how to extract tabular data from them. PDF tabular data may be of interest in various data extraction applications, including financial analytics (e.g. [1]), knowledge base construction (e.g. [7]), augmentation of Open Data [2] and Linked Open Data resources [19].

Table extraction as a part of *table understanding* [6] includes two phases: *table detection*, i. e. recovering the bounding box of a table in a document, and *table structure recognition*, i. e. recovering its rows, columns, and cells.

Many table extraction methods traditionally deal with only document images or plain-text [3,26]. Usually, they can also be applied to machine-readable PDF documents. However, the PDF-to-image conversion leads to the loss of valuable information, including text chunks and positions, font features, an order of appearance PDF instruction in a file, vector ruling lines, and positions of a drawing cursor. In comparison with images and plain-text, PDF is a richer representation of documents. PDF documents can contain machine-readable text as well as vector ruling lines. We expect that extracting tables from machine-readable PDF documents directly can provide more accurate results.

Several methods and tools for PDF table extraction are proposed in two last decades. Some of them are discussed in the recent surveys [2,3,9,10]. Ramel et al. [16] consider two techniques for detecting and recognizing tables from documents in an exchange format like PDF. The first is based on the analysis of ruling lines. The second is to analyze the arrangement of text components. Hassan et al. [8] expand these ideas for PDF table extraction. In the project TableSeer, Liu et al. [11] propose methods for detecting tables in PDF documents and extracting metadata (headers). They use text arrangement, fonts, whitespace, and keywords (e.g. "Table", "Figure"). Oro et al. [14] present PDF-TREX, a heuristic method where PDF table extraction is realized as building from content elements to tables in a bottom-up way.

Yildiz et al. [27] propose a heuristic method for PDF table extraction using pdftohtml[2] for generating its input. They also use the pdftohtml tool to prepare their input. However, this tool occasionally makes mistakes in combining text chunks, which are located too close to each other, thus the input can be corrupted. Nurminen [13] in his thesis describes comprehensive PDF table detection and structure recognition algorithms that have demonstrated high recall and precision on "ICDAR 2013 Table Competition" [4]. Some of them are implemented in Tabula[3], a tool for extracting tabular data from PDF. Rastan et al. [17] consider a framework for the end-to-end table processing including the task of table structure recognition. Moreover, Rastan et al. [18] suggest using an ad-hoc document analysis leading to better table extraction. Their wrapper able to detect features such as page columns, bullets, and numbering. Perez-Arriaga et al. [15] combines layout heuristics with a supervised machine learning method based on k-nearest neighbors to extract tables from untagged PDF documents. TAO, their system, promises to be an efficient, comprehensive and robust solu-

[2] http://pdftohtml.sourceforge.net.
[3] http://tabula.technology.

tion for both stages: table detection and cell structure recognition, that does not depend on fixed patterns or layouts of tables or documents.

We develop TabbyPDF, a novel web-based system for PDF table extraction from machine-readable untagged documents. This extends our previous work for table structure recognition [23]. The system exploits a set of customizable ad-hoc heuristics for table detection and cell structure reconstruction based on features of text and ruling lines presented in PDF documents. Most of them such as horizontal and vertical distances, fonts, and rulings are well known and used in the existing methods. Additionally, we propose to exploit the feature of appearance of text printing instruction in PDF files and positions of a drawing cursor. We also demonstrate experimental results based on the existing competition dataset, "ICDAR 2013 Table Competition". The standard metric F-score is 93.64% for the structure recognition phase and 83.18% for the table extraction with automatic table detection. The results are comparable with the state-of-the-art academic solutions.

2 PDF Table Extraction

The process of PDF table extraction involves the following phases:

1. *data preparation*, to recover text blocks presented words and ruling lines from instructions of a source PDF document;
2. *text line and paragraph extraction*, to recover text blocks presented lines and paragraphs;
3. *table detection*, to recover a bounding box of each table located on a page;
4. *table structure recognition*, to recover a cell structure of a detected table.

2.1 Data Preparation

We operate two kinds of objects:

– *text blocks* represent the text items: words, lines, paragraphs (including cell textual content);
– *ruling lines* serve as a representation of the graphic items: separator lines, and borders (including cell borders).

Text block consists of the following data:

– `bbox`—bonding box with four coordinates: `xl`—left, `yt`—top, `xr`—right, and `yb`—bottom, the x-coordinate increases from left to right, and y-coordinate increases from top to bottom;
– `font`—font with the attributes: `ff`—family (string value), `fs`—size in points, `fb`—bold or not, `fi`—italic or not;
– `order`—order of the appearance of text chunks in the source PDF file: `start`—index of a start text chunk in the order, and `end`—index of an end text chunk in the order.
– `ws`—width of a regular space in this font.

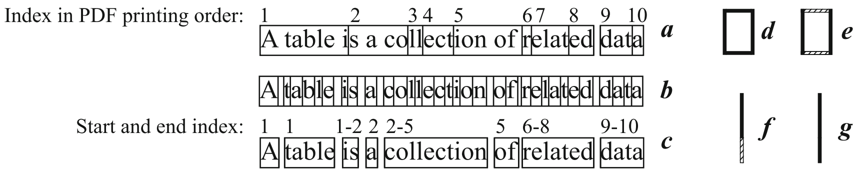


Fig. 1. Data gathered from an untagged PDF document: transforming text chunks (a) and positions (b) into text blocks representing words (c); splitting rectangles (d) into segments (f); merging segments (e) into ruling lines (g).

On this stage, each text block presented as a word is extracted from text chunks and positions of a PDF document, as shown in Fig. 1, *a*–*c*. We split all PDF text chunks (Fig. 1, *a*) into text positions (characters) (Fig. 1, *b*) and merge them into text blocks (words) with removing space characters and inducing the order of their appearance (Fig. 1, *c*).

Ruling line is characterized by four coordinates (`xl`—left, `yt`—top, `xr`—right, and `yb`—bottom) and `visibility` (visual or not). We separately recover both visual ruling lines presented borders and invisible ones that can usually represent traces of cursor motion (`moveto`, `lineto`). A ruling line can be originally presented by several PDF instructions for printing lines and rectangles. We split rectangles into segments by their bounds (Fig. 1, *d*–*e*). All segments are merged into ruling lines (Fig. 1, *f*–*g*).

2.2 Text Line and Paragraph Extraction

This stage aims to combine text blocks in order that each text block presents a textual content of one paragraph or one cell. We exclude preliminarily text blocks containing only itemization symbols (e.g. bullet, square). Typically, itemization symbols are visually detached from the rest of text chunks by long spaces. This lead to improperly recovered columns. We try to prevent these errors, eliminating these symbols from the further process, using the following assumption.

- H_1: a text block presenting a mark of an itemized list contains only one character of a specified set (bullet, square, etc.).

There are several ad-hoc heuristics we use to make a decision for each pair of text blocks to combine them or not. When a table is represented originally by a tagged form (e.g. as a table object in a Word-document) then PDF generators often store a logical reading order in printing text paragraph and cell content. We use the assumption that printing instructions forming a text inside each text or cell paragraph appear in the PDF file in the order that coincides with the human reading order of this text (Fig. 2, *a*–*b*). This feature can be especially useful to recover multi-line cell paragraphs without explicit borders. We also suppose that when ruling lines including invisible ones (cursor traces) are placed between two text blocks, then these text blocks belong to two separated paragraphs (Fig. 2, *c*).

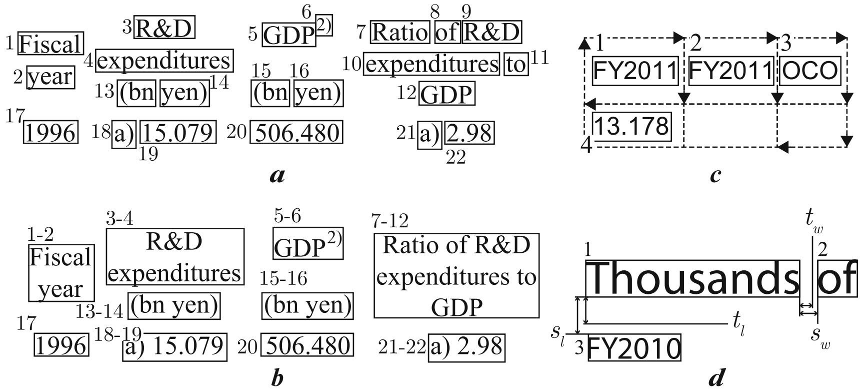


Fig. 2. Text block combination: words and their indices of the printing order (a) and cell paragraphs combined from the words (b); words are separated by the invisible ruling lines (cursor traces) (c); the word spacing (s_w) and line spacing (s_l) exceed the specified thresholds t_w and t_l respectively (d).

Two text blocks `tb1` and `tb2` can be combined into one `tb3` when they satisfy the specified assumption listed below:

- H_2, they are adjacent in the order of their appearance in the source PDF file: `tb1.order.end = tb2.order.start - 1`;
- H_3, there exist no ruling lines crossing `tb3.bbox`, the bounding box of the combined text block;
- H_4, they have identical fonts: `tb1.font = tb2.font` (including font family, size, bold, and italic attributes).

First, words are combined into text lines. The decision to combine a pair of text blocks `tb1` and `tb2` into one line `tb3` requires that they satisfy two additional assumptions:

- H_5, their word spacing, the horizontal distance between their bounding boxes (`tb2.xl - tb1.xr`), is less than a threshold depending on their width of space (`tb1.ws` or `tb2.ws`);
- H_6, their vertical projection intersection is more than a threshold.

Second, text lines are combined into paragraphs. Two text blocks `tb1` and `tb2` into one paragraph `tb3` when they satisfy two additional assumptions:

- H_7, their line spacing, the vertical distance between their bounding boxes (`tb2.yt - tb1.yb`), is less than a threshold depending on their font size (`tb1.fs` or `tb2.fs`);
- H_8, their horizontal projection intersection is more than a threshold.

The ideal case consists in that each combined text block is a textual content of one paragraph or cell, i. e. each non-empty cell contains only one text block.

2.3 Table Detection

TabbyPDF can detect both *bordered tables* (when borders presented explicitly as vector ruling lines) and *naked tables* (when their borders presented implicitly as invisible lines).

We assume that a bordered table is bounded by its outer borders represented by ruling lines. It also contains at least two rows and two columns completely bounded by ruling lines. To detect bordered tables, we select ruling lines that compose rectangular frames. When a frame contains or overlaps one or more others frames, then it is excluded from the processing. We search for groups of frames that intersect in their corner points. When a detected group of at least four frames composes at least two possible rows and two columns, then its outer ruling lines form a bounding box of a table.

Our approach to the naked table detection is based on separating a document into *table search areas*. A page can include one or more such areas (bounding boxes) where tables can potentially be placed. We try to detect some *table breaks* (e.g. section headings, keywords like "Table", bitmap figures, multi-line and stretched paragraphs, page headers and footers, as well as detected bordered tables) on each page. We assume that a table search area cannot include any of the detected breaks. Thus, we detect each area located between two neighbor breaks (Fig. 3, *e*).

To detect naked tables we use a bottom-up segmentation of a search area: from simpler layout items to more complicated ones (Fig. 4). This bottom-up approach is inherited from our previous work on table detection in metafiles [20].

Text blocks are grouped into one row when they are in the transitive closure of the following relation: there exists an intersection of y-projections of two text blocks. Therefore, each extracted row contains two or more blocks. Moreover, there are no intersected rows. A white-space surrounding text blocks inside a row is segmented by our algorithm [22]. We select *vertical gaps* from the obtained white-space segments. They can be considered as implicitly cell borders. We group neighbor rows agglutinated by their white-space into one *table region* (Fig. 4, *a*) when they satisfy the following assumptions: (i) the vertical distance between them does not exceed a specified threshold based on an inter-line interval used in its source document; (ii) the vertical gaps of both rows correlate with each other in their x-projections (i. e. there is at least one vertical gap of the lower row that is correlated with a gap of the upper row).

Detected neighbor regions can be combined into one table when their vertical white-space gaps also correlate with each other in their x-projections (Fig. 4, *b*). Additionally, we construct table columns using well-aligned blocks. When a detected table intersects a column of well-aligned blocks that goes beyond its horizontal borders, then we extend its upper and bottom borders by the column. Moreover, when there is a signal word indicating a start of a table (e.g. "Table"), then we extend the upper border of the table to the bottom border of the text block bounding this signal word.

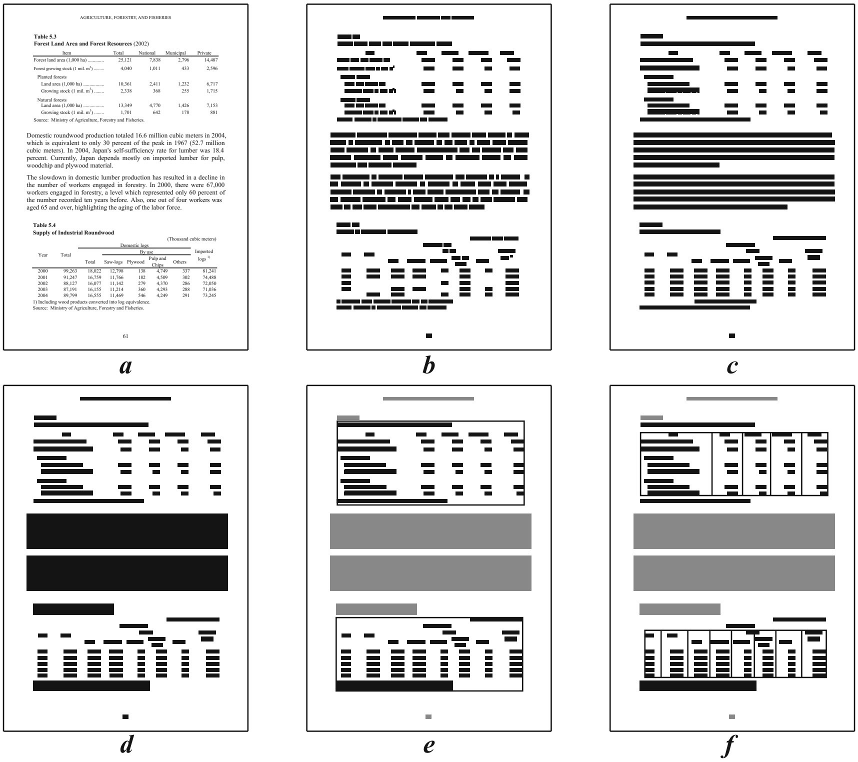


Fig. 3. The preliminary steps of the table detection: a source document (a); text blocks as words collected from "text positions" (b); text blocks as lines combined from words (c); text block as paragraphs combined from lines (d); table search areas located between table breaks (highlighted by gray color) (e); detected columns and tables (f).

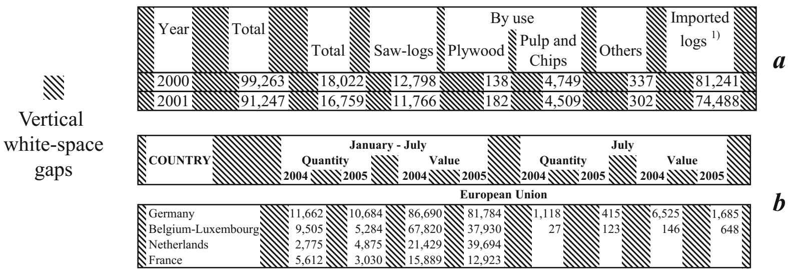


Fig. 4. The steps of the table detection in a table search area: combining neighbor rows agglutinated by their vertical white-space gaps into a region (a); combining neighbor regions agglutinated by their vertical white-space gaps into a table (b).

2.4 Table Structure Recognition

In this step, we construct rows and columns that constitute an arrangement of cells. The system provides two algorithms for slicing an inner space of a table into rows and columns. They both are presented more detail in our previous work [23].

The first (A_1) is based on the whitespace analysis. We use the algorithm [22] to recover horizontal and vertical gaps between text blocks. Each whitespace gap corresponds to a ruling. Thus, we try to recover all rulings, which separate cells in a table.

The second (A_2) is the analysis of connected text blocks. To generate columns, we first exclude each multi-column text block located in more than one column. We decide that a text block is multi-column when its horizontal projection intersects with the projections of two or more text blocks located in the same line. Each column is considered as an intersection of horizontal projections of one-column text blocks. Similarly, rows are constructed from vertical projections of one-row text blocks.

In this step, we also recover empty cells. Some of them can be erroneous, i. e. they absent in the source table. The system provides the ad-hoc heuristic to dispose of erroneous empty cells:

– H_9, *cell singleton*: if a column contains only one non-empty cell then the column is merged with the nearest column to the left.

3 Implementation

We develop a web-based prototype of TabbyPDF as a proof-of-concept for the presented approach. Its client-server architecture is shown in Fig. 5. The client part provides a web user interface built with Semantic UI[4]. It uses Mozilla PDF.js[5] tool for rendering PDF documents. The server part is a application based on Spring Framework[6]. It runs on Apache Tomcat[7] Java servlet container and stores data in the built-in database implemented by Apache Derby[8].

The client interacts with the server through REST-requests. Our RESTful API (Application Programming Interface) provides three requests:

– `/api/upload` POST-request uploads a PDF document specified in `file` parameter and returns positions of detected table bounding boxes in JSON format.
– `/api/extract` POST-request extracts tables from bounding boxes (positions in JSON format) specified in `data` parameter and returns extracted table cell structures in JSON format.

[4] https://semantic-ui.com.
[5] https://mozilla.github.io/pdf.js.
[6] https://spring.io.
[7] http://tomcat.apache.org.
[8] https://db.apache.org/derby.

– `/api/download/:id` GET-request builds ZIP-archive with extracted tables in HTML and CSV formats by an associated identifier provided in `id` parameter, and starts downloading this archive.

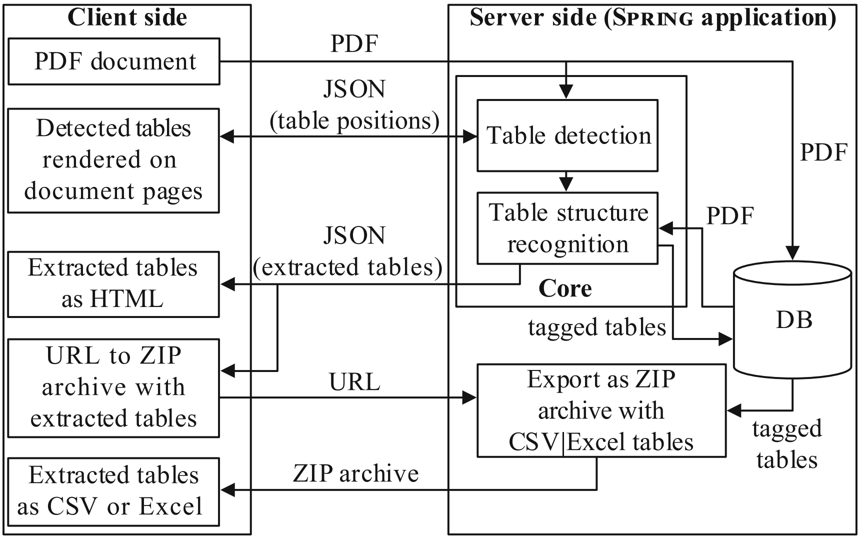


Fig. 5. The architecture of the web-based prototype of TABBYPDF.

The workflow includes the following steps. A user selects a PDF document on the client-side. The client uploads it to the server. The core detects tables in the PDF document on the server-side. The server responds to the user, returning detected table positions to the client. The client displays the detected table areas over rendered PDF pages for the user. The user can correct the table positions on the client-side. The client sends the user correction data to the server. The core extracts tables from the specified areas on the server-side. The server returns extracted tables to the client for viewing results by the user. The user can download all extracted data stored in CSV and HTML formats.

Our work is an ongoing open-source project published on GitHub[9]. Its current state is viewable online[10].

4 Performance Evaluation

To evaluate the performance of our system we use the methodology for algorithms for table understanding in PDF documents proposed in the paper [5].

[9] TABBYPDF core: https://github.com/cellsrg/tabbypdf.
TABBYPDF client: https://github.com/cellsrg/tabbypdf-front.
TABBYPDF server: https://github.com/cellsrg/tabbypdf-web.

[10] http://cells.icc.ru/pdfte.

We also use the existing competition dataset[11], “ICDAR 2013 Table Competition” [4]. It contains 156 tables in 67 PDF documents collected from EU and US government websites. The evaluation was performed automatically using Nurminen’s Python scripts[12] for comparing ground-truth and result files that implement this methodology with slight modifications.

The evaluated prototype of our system uses the iText[13] library for PDF interpretation to extract PDF objects from source documents and to generate initial text blocks and ruling lines. We use the threshold settings presented in the paper [23] as the C_2-configuration. The process of table extraction from all documents of this dataset performed by CPU (6M cache, up to 3.40 GHz) takes total 6342 ms: 3968 ms for the data preparation stage, 1563 ms for the table detection stage, and 810 ms for the cell structure recognition.

The experimental results are shown in Table 1. They are expressed by the standard metrics in the information extraction: *recall*, *precision*, and *F-score*. The evaluation methodology is presented in the papers [4,5] in detail. Briefly, these metrics are calculated (i) for both phases: the table detection and the structure recognition, separately, and (ii) for the fully automatic two-phase

Table 1. Experimental results

Scores	Detection phase	Recognition phase	Table extraction
Per-document averages			
Recall	0.8172	0.9233	0.8298
Precision	0.7605	0.9499	0.8339
F-score	0.7878	0.9364	0.8318

Table 2. Comparison with others in the table extraction

Tools	Per-document averages		*F-score*
	Recall	*Precision*	
FineReader	0.8835	0.8710	0.8772
OmniPage	0.8380	0.8460	0.8420
Nurminen [13]	0.8078	0.8693	0.8374
TabbyPDF	0.8298	0.8339	0.8318
Acrobat	0.7262	0.8159	0.7685
Nitro	0.6793	0.8459	0.7535
Silva [25]	0.7052	0.6874	0.6962
Yildiz et al. [27]	0.5951	0.5752	0.5850

[11] http://www.tamirhassan.com/dataset.html.
[12] http://tamirhassan.com/competition/dataset-tools.html.
[13] https://sourceforge.net/projects/itext.

process of the table extraction. The methodology first calculates *recall* and *precision* scores for each document separately and then calculates average values based on the document scores.

The experimental results are comparable with the state-of-the-art academic and some commercial solutions (Table 2). The comparison with others (Table 2) are based on the data presented in the paper [4].

5 Conclusion and Further Work

The main contribution of this work consists in demonstrating experimentally new possibilities of using the order of text printing commands as well as the cursor motion commands presented in PDF files to efficiently detect and extract untagged tables. We have formulated a set of valuable ad-hoc heuristics using these possibilities that provide combining words into text and cell paragraphs. This often allows to reduce errors of the table extraction.

The main advantage of our approach is that it can be quickly adapted to the table extraction from various domain-specific PDF documents (such as financial statements, business credit assessments, material safety data sheets, etc.) with a rich tabular content. Additional ad-hoc heuristics can be handcrafted and tuned for the target domain. This does not require to prepare training datasets that can be a costly process. The main disadvantage of this approach consists in that it is unsuitable when a PDF document is rasterized or contains glyphs without appropriate mappings to a standard encoding (e.g. Unicode).

The web-based prototype of TabbyPDF enables both automatic table detection and automatic cell structure recognition. It exports extracted tables in the editable format, CSV or HTML. Its output can be used as input in our rule-based spreadsheet data extraction system for transforming extracted arbitrary tables to relational ones [21,24].

The further work is in progress on expanding the set of ad-hoc heuristics. The table extraction based on PDF content analysis requires an advance in document layout analysis (e.g. page header and footer detection; page column segmentation; title, section, and paragraph detection). We believe the involvement of the additional document layout analysis will allow improving our system. We also expect an advancement in involving ruling lines represented both PDF rendering instructions and ASCII pseudographics. This will provide extracting bordered tables more accurately.

Acknowledgments. This work is supported by the Russian Foundation for Basic Research (grants 18-07-00758 and 17-47-380007). The prototype of TabbyPDF is deployed on resources of the Shared Equipment Center of Integrated Information and Computing Network for Irkutsk Research and Educational Complex (http://net.icc.ru).

References

1. Burdick, D., et al.: Financial analytics from public data. In: Proceedings of the International Workshop on Data Science for Macro-Modeling, DSMM 2014, pp. 4:1–4:6 (2014). https://doi.org/10.1145/2630729.2630742
2. Corrêa, A.S., Zander, P.O.: Unleashing tabular content to open data: a survey on PDF table extraction methods and tools. In: Proceedings of 18th International Conference on Digital Government Research, pp. 54–63 (2017). https://doi.org/10.1145/3085228.3085278
3. Coüasnon, B., Lemaitre, A.: Recognition of tables and forms. In: Handbook of Document Image Processing and Recognition, pp. 647–677 (2014). https://doi.org/10.1007/978-0-85729-859-1_20
4. Göbel, M., Hassan, T., Oro, E., Orsi, G.: ICDAR 2013 table competition. In: Proceedings of 12th International Conference on Document Analysis and Recognition, pp. 1449–1453 (2013)
5. Göbel, M., Hassan, T., Oro, E., Orsi, G.: A methodology for evaluating algorithms for table understanding in PDF documents. In: Proceedings of 2012 ACM Symposium on Document Engineering, pp. 45–48 (2012). https://doi.org/10.1145/2361354.2361365
6. Göbel, M., Hassan, T., Oro, E., Orsi, G., Rastan, R.: Table modelling, extraction and processing. In: Proceedings of 2016 ACM Symposium on Document Engineering, pp. 1–2 (2016). https://doi.org/10.1145/2960811.2967173
7. Govindaraju, V., Zhang, C., Ré, C.: Understanding tables in context using standard NLP toolkits. In: Proceedings of 51st Annual Meeting of the Association for Computational Linguistics, pp. 658–664 (2013)
8. Hassan, T., Baumgartner, R.: Table recognition and understanding from PDF files. In: Proceedings of 9th International Conference on Document Analysis and Recognition, vol. 02, pp. 1143–1147 (2007). http://dl.acm.org/citation.cfm?id=1304596.1304833
9. Hu, J., Liu, Y.: Analysis of documents born digital. In: Doermann, D., Tombre, K. (eds.) Handbook of Document Image Processing and Recognition, pp. 775–804. Springer, London (2014). https://doi.org/10.1007/978-0-85729-859-1_26
10. Khusro, S., Latif, A., Ullah, I.: On methods and tools of table detection, extraction and annotation in PDF documents. J. Inf. Sci. **41**(1), 41–57 (2015). https://doi.org/10.1177/0165551514551903
11. Liu, Y., Bai, K., Mitra, P., Giles, C.L.: TableSeer: automatic table metadata extraction and searching in digital libraries. In: Proceedings of 7th ACM/IEEE Joint Conference on Digital Libraries, pp. 91–100 (2007). https://doi.org/10.1145/1255175.1255193
12. Nganji, J.T.: The portable document format (PDF) accessibility practice of four journal publishers. Libr. Inf. Sci. Res. **37**, 254–262 (2015). http://www.sciencedirect.com/science/article/pii/S0740818815000134
13. Nurminen, A.: Algorithmic extraction of data in tables in PDF documents. Master's thesis, Tampere University of Technology, Tampere, Finland (2013)
14. Oro, E., Ruffolo, M.: PDF-TREX: an approach for recognizing and extracting tables from PDF documents. In: Proceedings of 10th International Conference on Document Analysis and Recognition, pp. 906–910 (2009)
15. Perez-Arriaga, M.O., Estrada, T., Abad-Mota, S.: TAO: system for table detection and extraction from PDF documents. In: Proceedings of 29th International Florida Artificial Intelligence Research Society Conference, pp. 591–596 (2016)

16. Ramel, J.Y., Crucianu, M., Vincent, N., Faure, C.: Detection, extraction and representation of tables. In: Proceedings of 7th International Conference on Document Analysis and Recognition, vol. 1, pp. 374–378 (2003)
17. Rastan, R., Paik, H.Y., Shepherd, J.: TEXUS: a task-based approach for table extraction and understanding. In: Proceedings of 2015 ACM Symposium on Document Engineering, pp. 25–34 (2015). https://doi.org/10.1145/2682571.2797069
18. Rastan, R., Paik, H.Y., Shepherd, J.: A PDF wrapper for table processing. In: Proceedings of 2016 ACM Symposium on Document Engineering, pp. 115–118 (2016). https://doi.org/10.1145/2960811.2967162
19. Sabol, V., Tschinkel, G., Veas, E., Hoefler, P., Mutlu, B., Granitzer, M.: Discovery and visual analysis of linked data for humans. In: Mika, P., et al. (eds.) ISWC 2014. LNCS, vol. 8796, pp. 309–324. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11964-9_20
20. Shigarov, A., Bychkov, I., Ruzhnikov, G., Khmel'nov, A.: A method of table detection in metafiles. Pattern Recognit. Image Anal. **19**(4), 693–697 (2009). https://doi.org/10.1134/S1054661809040191
21. Shigarov, A.: Table understanding using a rule engine. Expert. Syst. Appl. **42**(2), 929–937 (2015)
22. Shigarov, A., Fedorov, R.: Simple algorithm page layout analysis. Pattern Recognit. Image Anal. **21**(2), 324–327 (2011). https://doi.org/10.1134/S1054661811021008
23. Shigarov, A., Mikhailov, A., Altaev, A.: Configurable table structure recognition in untagged PDF documents. In: Proceedings of 2016 ACM Symposium on Document Engineering, pp. 119–122 (2016). https://doi.org/10.1145/2960811.2967152
24. Shigarov, A.O., Mikhailov, A.A.: Rule-based spreadsheet data transformation from arbitrary to relational tables. Inf. Syst. **71**, 123–136 (2017). https://doi.org/10.1016/j.is.2017.08.004
25. e Silva, A.C.: Parts that add up to a whole: a framework for the analysis of tables. Ph.D. thesis, University of Edinburgh, Tampere, Finland (2010)
26. e Silva, A.C., Jorge, A.M., Torgo, L.: Design of an end-to-end method to extract information from tables. Int. J. Doc. Anal. Recognit. (IJDAR) **8**(2), 144–171 (2006)
27. Yildiz, B., Kaiser, K., Miksch, S.: pdf2table: a method to extract table information from PDF files. In: Proceedings of 2nd Indian International Conference on Artificial Intelligence, Pune, India, pp. 1773–1785 (2005)

Modification of Parallelization for Fast Sort Algorithm

Zbigniew Marszałek(✉)

Institute of Mathematics, Silesian University of Technology, ul. Kaszubska 23, 44-100 Gliwice, Poland
Zbigniew.Marszalek@polsl.pl

Abstract. One of the most important issues in NoSQL databases is to develop applications and facilitates for the parallel processing in information systems. In the work the author presents some improvements for the parallel algorithm for merging strings and use this algorithm to sort large data sets. Tested sorting for a parallel merging algorithm confirms the reduction of the time complexity and improved stability of the algorithm.

Keywords: Parallel algorithm · Data sorting · Data mining Analysis of computer algorithms

1 Introduction

Modern computers process large amounts of data, while for processing applications make use of machines equipped with a number of processors and massive data storage systems. As always in such situations, it is important to perform computations in parallel by splitting tasks on separate processors for better efficiency. The problem of the distribution of sorted collections using various sorting algorithms is described in [1–4], where authors proposed some changes in flip state procedures or direct methods which work on parallel sequences. Various derivatives of sorting procedures find their applications in data mining [5, 6], image processing [7, 8], medical expertise automated support [9], mobile computing [10], and positioning of object in global systems from large data sets [11–13]. The aim of this work is to present the possibility of using a larger number of processors at each stage of the merge strings than it was done in [14]. By using a new parallel merge algorithm of strings increased the number of processors in each step twice.

1.1 Related Work

The problem of parallel computing and in particular the parallel execution of calculations in sorting algorithms was presented in [2, 4]. Continued theoretical work on the search for improved sorting algorithm with the lowest computational complexity settled in several applications. Bag of features methods always use a sequence of descriptors to enhance selection of objects from initial collection, these can be images [15], dictionaries [16], and models oriented on processing in NoSQL systems [17].

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 270–278, 2018.
https://doi.org/10.1007/978-3-319-99972-2_21

The proposed algorithms [11, 14, 18, 19] are trying to fill the resulting gap in practical research on parallelization of sorting procedures. In these articles I have started the following improvements to faster and more efficient sorting in a parallel mode, where the improvements were proposed for selections of elements, data transmission and lower complexity. The algorithm proposed in this paper, is $\frac{3}{5}$ the time constant in relation to the previously proposed parallel fast sort algorithm [14]. This result has been achieved through the use of devoted parallel method of merging numerical strings. The tests fully confirmed the theoretical results. This allows to additionally reduce the constant in relation to the previous algorithm with time complexity $O(n)$.

2 Big Data Sets and Parallel Algorithms

In theoretical studies on the computational complexity of parallel algorithms we use PRAM (Parallel Random Access Machine) machine model. PRAM is an extension and generalization of the RAM model, which use multiple processors. Machine-PRAM we can define as a system consisting of n processors $P_i, i = 0, \ldots, n-1$ and shared memory. Each processor has a specific index and the define instructions that it can perform. It is assumed that each instruction executes in unit time. The time complexity of the algorithm can be defined as the number of statements executed during the algorithm.

The architecture of modern processors allows to perform in a parallel way multiple processes. For the analysis of algorithms running on modern computers we use the theoretical multiprocessor machine model PRAM (Parallel Random Access Machine). Due to the access of the processors to read and write into memory we can talk about four PRAM machines:

- Concurrent Read Concurrent Write (CRCW)
- Exclusive Read Concurrent Write (ERCW)
- Concurrent Read Exclusive Write (CREW)
- Exclusive Read Exclusive Write (EREW)

Big Data describes an information management strategy that includes and integrates many new types of data and data management alongside traditional data. The defining processing capabilities for big data architecture are to meet the volume, velocity, variety, and value requirements. While many of the techniques to process and analyze these data types have existed constantly working on the cost reduction and performance acceleration application.

Traditionally, information in the NoSQL database is stored in the form of records of fixed or variable length. When retrieving records in memory, the entries are grouped in small blocks, which enables the parallelization of queries of the calculation process. As usual in such situations, the sorting algorithms of data stets play a big role. Parallel the sort algorithm to speed up the application find information.

The modification of parallelization for fast sort algorithm describes how to split the sorting processes between independently running processors when using a new parallel algorithm for merging two strings.

3 Parallel Fast Sort Algorithm

Almost everyone is aware that Big Data are currently being used to analyze the market and make the right decisions. For example, applications enabling better customer experiences are often powered by smart devices and enable the ability to respond in the moment to customer actions. Processing large sets requires applications that use the possibility of parallel calculations on modern computers. A special role in the sorting of the data sets play sort algorithms by merging strings. I will now present the modification of the parallel fast sorting algorithm PFSA by merging tree strings [14]. The modification is based on the use of parallel merge algorithm of two strings that will reduce the time complexity of sorting algorithm using multiple processors.

3.1 Parallel Sort Algorithm

Let's leave out how to use a parallel merge algorithm of two numeric strings in Fast Sort Algorithm [14] aside and focus on proposed improvements. Merged first two strings are saved in a temporary array. Left third string should be rewritten into the temporary array. In the first phase of the merger of three strings, when using a parallel merging algorithm strings can be divided into three processors working independently. Assign tasks to the processors so that each processor will save the information in this section is this part of the temporary array that is allocated to write for that particular processor. Figure 1 illustrates how centrally of strings processors from the array input and the temporary array in the first iteration.

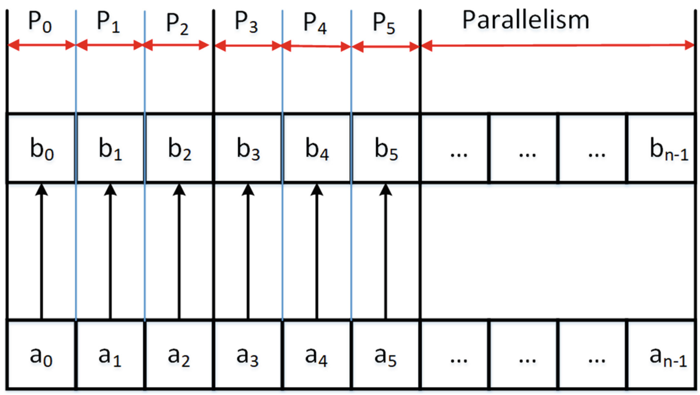


Fig. 1. Parallelism merge strings in the array b.

In the next step the algorithm merges the strings into an array from the input with the parallel merging algorithm two strings. The method in which tasks are divided between the processors was presented in Fig. 2.

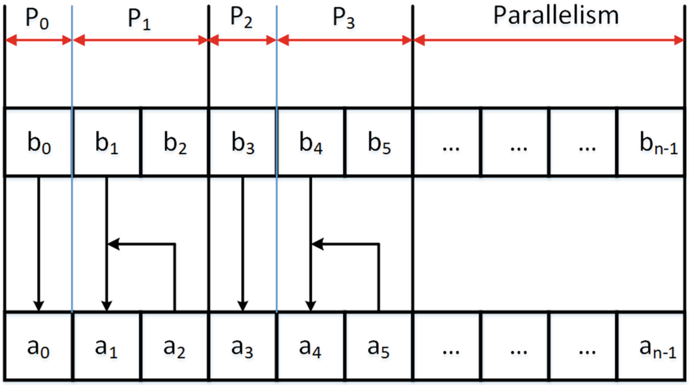


Fig. 2. Parallelism merge strings the input array.

Theorem 1. Computational complexity of the modified sorting algorithm is:

$$T_{max} = \frac{3}{2}n - \frac{3}{2}. \tag{1}$$

Proof. The proof will be carried out for $n = 3^k$, $k = 1, 2, \ldots$.

Merging strings $x_1 \leq \cdots \leq x_t$ and $y_1 \leq \cdots \leq y_t$ into the sequence $z_1 \leq \cdots \leq z_{2t}$, using a parallel merge algorithm with two processors running independently, we do our best t comparison elements of sequences X and Y on each processor.

In each iteration $t = 1, \ldots, k$, the sorting algorithm performs two actions. The first step is the merger of two parallel strings about 3^{t-1} elements and saving the third for 3^{t-1} of elements in the temporary array. When performing this operation, each independently operating processors performs no more than 3^{t-1} comparisons of elements in strings of numbers. The second step is the concatenation of strings of length 3^{t-1} and $2 \cdot 3^{t-1}$ and stores the result in the input array. A parallel merging algorithm will perform then do not more than $2 \cdot 3^{t-1}$ comparisons of elements of the joint strings. By summing the number of comparisons that are performed in both steps we obtained an upper estimate of working time of each iteration.

$$3^{t-1} + 2 \cdot 3^{t-1} \tag{2}$$

By aggregating all the operations performed by the processors get

$$\begin{aligned} &\sum_{t=1}^{k} \left(3^{t-1} + 2 \cdot 3^{t-1}\right) = 3 \cdot \sum_{t=1}^{k} 3^{t-1} \\ &= 3\left(1 + 3 + \cdots + 3^{k-1}\right) - 2k = 3 \cdot \left(\frac{3^k - 1}{2}\right) \\ &= \frac{3}{2}n - \frac{3}{2} \end{aligned} \tag{3}$$

which was to prove.

In the implementation in Visual Studio Ultimate 2013 of the presented sorting algorithm, the C# class System.Threading.Tasks was used. This class has a parallel for loop, which is not an iterative in classic understanding, but an iteration that assigns tasks to the processes that are called for each of the processors. Subsequent processes receive the identifier of consecutive natural numbers, starting with zero and ending with one less than the number of processes. According to the object-oriented programming principle, variables declared inside this loop are only available for a given thread, and arrays and variables declared outside the parallel loop are available for all processes. Synchronization of all processes occurs after the parallel loop ends.

3.2 Experimental Study of the Parallel Sort Algorithm

Benchmark tests of the newly proposed modification of polarization for fast sort method were taken for 100, …, 100000000 elements on the input. The results are presented in tables and discussed in the following figures. The purpose of the analysis and comparison is to verify how the newly proposed parallel processing can speed up sorting of data sets. Presented results are averaged for 100 sorting samples. Studies were made in C# in Visual Studio Ultimate 2013 on Microsoft Sever 2012 (Tables 1 and 2).

Table 1. Benchmark tests in [ms].

The number of elements in the sample	One – processor	Two – processors	Four – processors	Eight – processors
100	1	1	1	1
1 000	1	1	1	1
10 000	7	5	4	3
100 000	75	42	29	24
1 000 000	760	399	276	212
10 000 000	8475	4524	3134	2228
100 000 000	9369	5031	33746	22865

A comparison of speed of proposed modified method for parallel sorting of data sets using 1, 2, 4 and 8 processors is shown in Figs. 3 and 4.

Tables 3 and 4 shows the coefficient of variation of the modified method parallel merge numeric strings.

The tests carried out demonstrate the stability of the presented algorithm for any dataset dimension. A parallel sorting method for merging three numerical strings can be used in NoSQL databases and in sorting small data sets.

3.3 Analysis of Sorting Time and Compression

Consider a basis for the run-time comparison of the algorithm on a single processor and see how much faster the calculation will take with more processors. The acceleration result is shown in Figs. 5 and 6.

Table 2. Benchmarks test in [ti].

The number of elements in the sample	One – processor	Two – processors	Four – processors	Eight – processors
100	383	346	296	210
1 000	1064	819	742	671
10 000	10553	6673	5607	4975
100 000	116148	65113	45037	35649
1 000 000	1184754	621992	430354	324575
10 000 000	13209619	7050969	4884816	3496753
100 000 000	146464059	78416650	52597832	36866278

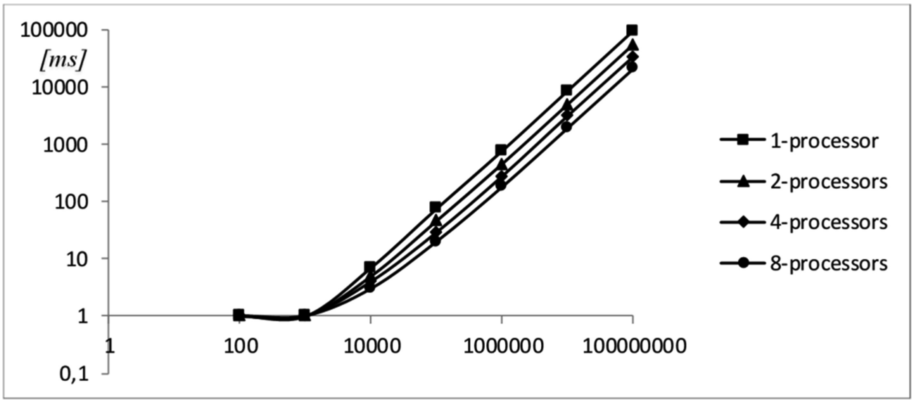


Fig. 3. Benchmarks test in [ms]

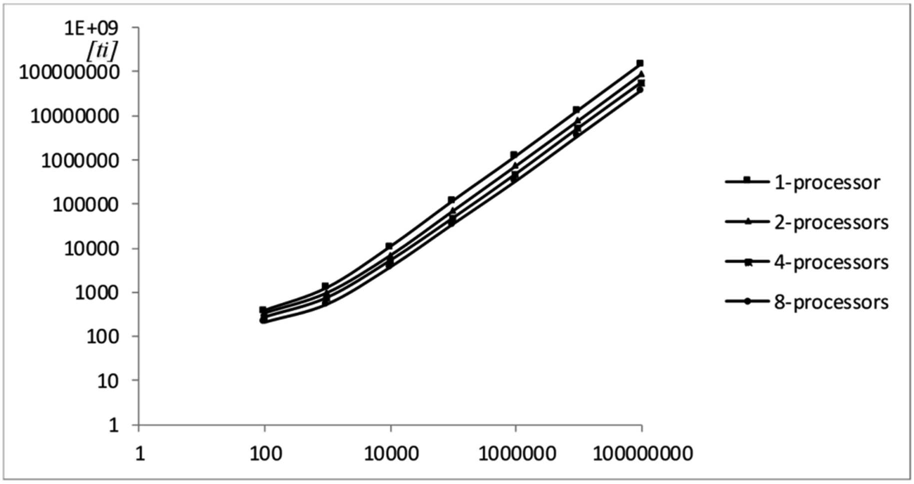


Fig. 4. Benchmarks test in [ti]

Table 3. The obtained coefficients of variation for the presented method [ms].

The number of elements in the sample	One – processor	Two – processors	Four – processors	Eight – processors
100	0.1425638	0.2055871	0.1183261	0.1020143
1 000	0.1263521	0.1404761	0.1224651	0.0815983
10 000	0.2057423	0.1636634	0.1185492	0.1781741
100 000	0.1871422	0.1842080	0.1001103	0.1184508
1 000 000	0.0993884	0.1059641	0.0990169	0.0751336
10 000 000	0.1052747	0.1091599	0.0661096	0.0748716
100 000 000	0.108490	0.1165212	0.1134420	0.0825547

Table 4. The obtained coefficients of variation for the presented method [ti].

The number of elements in the sample	One – processor	Two – processors	Four – processors	Eight – processors
100	0.147538	0.2086850	0.1044300	0.0914580
1 000	0.1232722	0.0922867	0.1191738	0.0434745
10 000	0.1773820	0.0997180	0.0781412	0.0809674
100 000	0.1875816	0.1877015	0.0952957	0.1193617
1 000 000	0.0993503	0.1062244	0.0990616	0.0750950
10 000 000	0.1052772	0.1092027	0.0660495	0.0748664
100 000 000	0.1084909	0.1165216	0.1134390	0.0825522

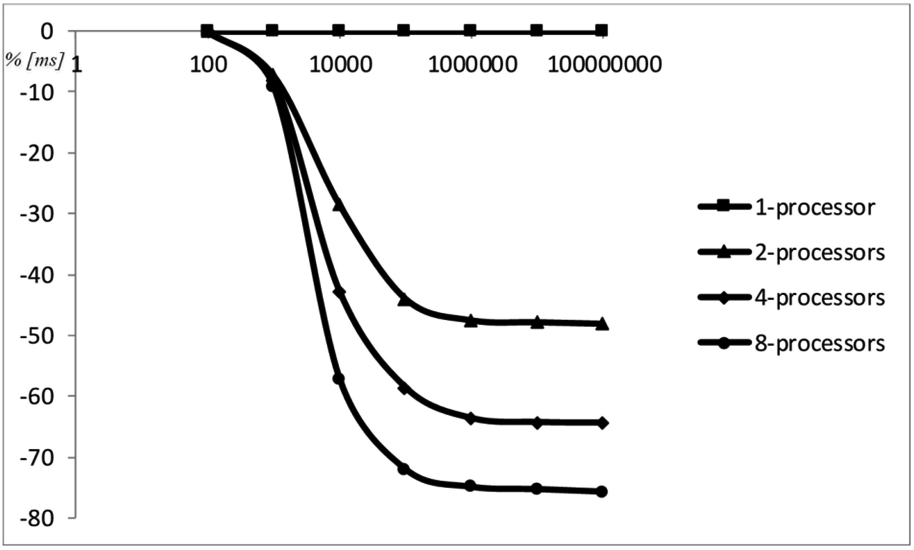


Fig. 5. The acceleration result [ms].

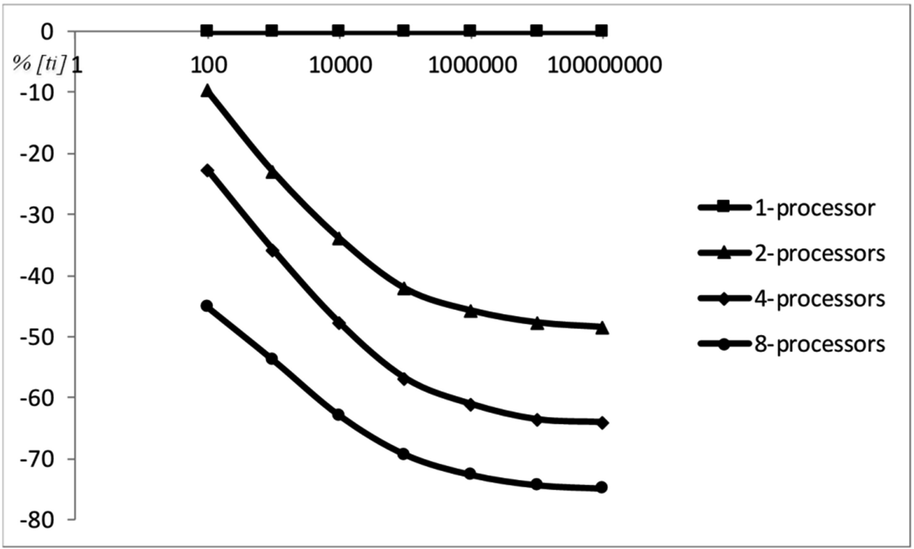


Fig. 6. The acceleration result [ti].

From the results in Fig. 5 we see that each new processor additionally boosts the sorting performance by decreasing sorting time and necessary operations, and therefore increases the efficiency of the method. Figure 6 shows that the application of additional processors improves the method by about 80%, for large data set, what improves sorting for large data sets.

4 Final Remarks

The work presents the use of a parallel merge algorithm to merge numeric strings to speed up the sorting of large data sets in parallel by the proposed Fast Sort Algorithm. The method presented in this article effectively organizes large amounts of data using a number of processors. The tests fully confirmed the theoretical computational complexity and the stability of the algorithm. The algorithm can be applied to sorting data sets using a limited number of processors.

References

1. Shi, H., Schaeffer, J.: Parallel sorting by regular sampling. J. Parallel Distrib. Comput. **14**(4), 361–372 (1992)
2. Preparata, F.P.: New parallel-sorting schemes. IEEE Trans. Comput. **7**, 669–673 (1978)
3. Rajasekaran, S., Reif, J.H.: Optimal and sublogarithmic time randomized parallel sorting algorithms. SIAM J. Comput. **18**(3), 594–607 (1989)
4. Hirschberg, D.S.: Fast parallel sorting algorithms. Commun. ACM **21**(8), 657–661 (1978)

5. Artiemjew, P.: Ensemble of classifiers based on simple granules of knowledge. In: Damaševičius, R., Mikašytė, V. (eds.) ICIST 2017. CCIS, vol. 756, pp. 343–350. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67642-5_28
6. Artiemjew, P., Nowak, B.A., Polkowski, L.T.: A new classifier based on the dual indiscernibility matrix. In: Dregvaite, G., Damasevicius, R. (eds.) ICIST 2016. CCIS, vol. 639, pp. 380–391. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46254-7_30
7. Woźniak, M., Połap, D.: Adaptive neuro-heuristic hybrid model for fruit peel defects detection. Neural Netw. **98**, 16–33 (2018)
8. Woźniak, M., Połap, D.: Bio-inspired methods modeled for respiratory disease detection from medical images. Swarm Evol. Comput. (2018)
9. Lauraitis, A., Maskeliūnas, R., Damaševičius, R.: ANN and Fuzzy logic based model to evaluate huntington disease symptoms. J. Healthcare Eng. 2018 (2018)
10. Maskeliūnas, R., Damaševičius, R.: Developing a calorie counter fitness app for smartphones. Inf. Technol. Sci. **724**, 23 (2018)
11. Włodarczyk-Sielicka, M., Wawrzyniak, N.: Problem of bathymetric big data interpolation for inland mobile navigation system. In: Damaševičius, R., Mikašytė, V. (eds.) ICIST 2017. CCIS, vol. 756, pp. 611–621. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67642-5_51
12. Wlodarczyk-Sielicka, M., Bodus-Olkowska, I.: Simplification methods for line and polygon features in mobile navigation systems for inland waters. Sci. J. Marit. Univ. Szczecin **50**, 105–111 (2017)
13. Wlodarczyk-Sielicka, M., Stateczny, A.: General concept of reduction process for big data obtained by interferometric methods. In: 2017 18th International Radar Symposium (IRS), pp. 1–10. IEEE (2017)
14. Marszałek, Z.: Parallelization of modified merge sort algorithm. Symmetry **9**(9), 176 (2017)
15. Gabryel, M.: The bag-of-words methods with pareto-fronts for similar image retrieval. In: Damaševičius, R., Mikašytė, V. (eds.) ICIST 2017. CCIS, vol. 756, pp. 374–384. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67642-5_31
16. Gabryel, M., Capizzi, G.: The bag-of-words method with dictionary analysis by evolutionary algorithm. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2017. LNCS (LNAI), vol. 10246, pp. 43–51. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59060-8_5
17. Gabryel, M.: A bag-of-features algorithm for applications using a NoSQL database. In: Dregvaite, G., Damasevicius, R. (eds.) ICIST 2016. CCIS, vol. 639, pp. 332–343. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46254-7_26
18. Marszałek, Z.: Performance tests on merge sort and recursive merge sort for big data processing. Technical Sciences **21**(1), 19–35 (2018)
19. Marszałek, Z.: Novel recursive fast sort algorithm. In: Dregvaite, G., Damasevicius, R. (eds.) ICIST 2016. CCIS, vol. 639, pp. 344–355. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46254-7_27

Text Semantics and Layout Defects Detection in Android Apps Using Dynamic Execution and Screenshot Analysis

Šarūnas Packevičius(✉), Dominykas Barisas, Andrej Ušaniov, Evaldas Guogis, and Eduardas Bareiša

Software Engineering Department, Kaunas University of Technology, Kaunas, Lithuania
sarunas.packevicius@ktu.edu

Abstract. The paper presents classification of the text defects. It provides a list of user interface text defects and the method based on static/dynamic code analysis for detecting defects in Android applications. This paper proposes a list of static analysis rules for detecting every defect and the tool model implementing those rules. The method and the tool are based on the application of multiple Android application emulators, execution of the application through certain execution paths on multiple hardware and software configurations while taking application screen-shots. The defects are identified by running analysis rules on each taken screen-shot and searching for defect patterns. The results are presented by testing sample Android application.

1 Introduction

The current Android applications market demands developers to build applications that can work on a wide array of mobile devices that have various hardware and software configurations. One of the common problems is a clipped text that does not fit on the applications window. This defect can become evident when an application is localized to other language and text messages become longer than it was originally anticipated. Those texts usually do get clipped and contain three-dot endings. Some of the other defects are: a text is too small to see or too large and it takes too much of the screen space on some devices.

The problems become evident when developers create applications on one primary development device and make its appearance consistent and refined. However, when the application is used on other devices (for example, on the ones having smaller screens) with a different configuration (for example, the application is internationalized to a Hungarian language, in contrast to the primary language it was developed, i.e. English), the text gets clipped and most of it is replaced by dots (…) as the text becomes twice longer than originally intended. To detect those defects, testers would have to test every application functionality in all application languages and on a wide array of device configurations.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 279–292, 2018.
https://doi.org/10.1007/978-3-319-99972-2_22

2 Related Work

In the previous paper [1] we presented the initial method for detecting various types of user interface defects on mobile devices. The focus in this paper lies in text defect types, by further extending the text defects classification, proposing the text defect detection technique.

There are several works that address graphical user interface defects detection. They fall into two categories: detecting defects from the application images and detecting defects from the application source code.

The defects detection by analysing an application code falls into a category of static code analysis. However, these methods search for graphical user interface defects instead of coding errors [2]. Lelli et al. presented the method [3] for detecting defects by analysing an application source code and analysing event listeners.

Moran et al. presented the method [4] for detecting user interface defects by analysing application mock-ups and screenshot images. This method relies on having application mock-ups that are not always available or must perfectly match a developed application. Chang et al. presented the method for user interface testing using computer vision [5, 6]. The image recognition was used for detecting controls in an application interface and using them in automated testing scripts. Baek et al. presented the model based method [7] that detects user interface defects by comparing user interface to its model.

3 The Proposed Method

For the proposed method we combine two approaches for defects detection. We use application code and application screenshots to identify defects and pinpoint their location in situations where a user would fail. In addition, the application source code and corresponding assets are used as an application model that defines what kind of messages should be visible on the screen.

The method description is presented in further sections:

1. Defining defect classes – defect types are identified.
2. Rules set – a set of algorithms capable of detecting defects of each identified type.
3. Situation modelling – the method that must be automated by a tool, for generating situations where defects could arise.

The method is based on the principle of a source code static analysis: to gather data to analyse, run a set of rules for detecting each type of defect, and present findings.

The common static analysis tools analyse application source code. We propose to analyse the application screenshots in a similar manner. We define a set of defects and rules for detecting those defects in the application screenshots.

3.1 Classification of the User-Interface Text Defects

There are several defects classification methods such as presented in [8–10] that identify common defect types, such as data, interface, logic, function, and documentation. The interface defect types are divided into user-interface and component-interface defect types. Lelli et al. [3] provided the scheme for classification of user-interface defects. Under the presented scheme defects fall into two categories: GUI Structure and Aesthetics, and Data Presentation. Based on the defect classifications [1, 4, 11] we concentrate on user-interface defects in the presentation category. Furthermore, we provide the list of defect types and methods for automatically detecting defects that are related to textual defect sub-types. Further on, we divide text defects into two categories:

1. Presentation – message presentation. The questions “is it visible”, “is it unobscured”, etc. are raised.
2. Semantics – defects that are related to a text message itself. This refers to an incorrect message, a misleading message, difficult to understand message, etc.

Presentation defects are related to applications text visibility on the screen. The main aspects are correct location, visibility, clashing with the background. The list is presented in Table 1.

Table 1. Text presentation defects.

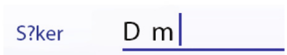
No	Type	Description	Example
TP1	Text placement	Text and surrounding controls are not aligned (text field and its label are not aligned vertically)	
TS1	Font sizes	The sizes of the text and surrounding controls mismatch (text size of the text field and its label are too different)	
TS2	Unreadable text	Text is too small to read on a device (with small physical screens and low resolution)	
TB1	Clashing background	Text colour clashes with a background image or colour, making the text difficult or impossible to read	
TC1	Partial text	Text is obscured by other controls or screen edges	
TC2	Clipped text	Text is too long to be presented in an allocated screen area, text is truncated by an user interface toolkit and is replaced by “…”	
TE1	Wrong encoding	Messages lack some characters, wrong characters are displayed, characters are replaced by question marks or a space character	
TM1	Missing text	Text is not visible	

Table 2. Text semantics defects.

No	Type	Description	Example
SD1	Synonyms	Different words are used to describe the same object	*The schedule* is assigned. *The week program* is not set.
SD2	Wrong terminology	Inconsistent terminology, different terms for the same concept, the same name for different concepts is used	*Loyalty card* is expired. (instead of a "*gift card*")
SD3	Unclear terminology	Ambiguous terms, terms for different concepts overlap in the meaning, too similar concepts, the same name used for different concepts	Heater is off. (meaning "electric one") Heater is on. (meaning "thermostat")
SD4	Jargon	Technical jargon, programmer jargon	Establishing connection to remote web socket.
SS1	Bad text style	Inconsistent writing style, misspelled words, incorrect grammar and/or punctuation	press und hold button for fvie sek chek red LED.
SS2	Too much text	Messages are too long to read	No program is assigned to this zone. To program this zone, you should either create a new program or select an existing program from the list below.
SS3	Vague error messages	Messages do not provide any meaningful information	Connection was reset by peer
SS4	Untactful messages	Offensive, impolite messages	Error!!! Invalid input fields. Enter correct data!!!
SS5	Misleading text	Erroneous messages	Device firmware does not support this method, please use the first method
SM1	Meaningless text	Text makes sense in an isolation but is meaningless in a GUI	
SU1	Long instructions	Instructions are too long, unstructured, not illustrated	Press and hold [WiFi] button for 15 seconds until red light starts blinking and then starts blinking very fast. Then click "Next". If LED only blinks slowly, your WiFi Smart Plug does not yet support WPS mode, please try AP registration option.
SU2	Too difficult to understand the text	Text writing style makes it difficult to understand	As many automated test input generation tools for Android need to instrument the system or the app, they cannot be used in some scenarios such as compatibility testing and malware analysis.
SU3	"…" misuse	The command with "…" should denote a link to additional window but it does not lead to it	Save… Exit…
SL1	Missing translations	Not all application messages are translated, translation placeholders are visible	error_message_no_wifi
SL2	Wrong language	The application shows a message in a wrong language, shows messages in mixed languages at once	You could reset device to factory defaults and try again. Trykk og hold nede [+], [-] and [OK] knapper på gulvvarmetermostaten i 5 sekunder.

The sematic defects are related to meanings of the messages. The main aspects are vague messages, too long messages, technical jargon, and unclear terminology. The list is provided in Table 2.

3.2 Detection of the Defects

We provide rules for detecting each defect declared in Sect. 1. The rules are executed on each application screen-shot, they apply data extracted from the application screen-shots, source code and runtime data. Each rule is targeted for identifying only one possible defect type. The rules use data from 3 different sources:

- S Screen-shot – defects are identified by comparing reference texts with texts extracted from screen-shots. A single window screen-shot is analysed in isolation.
- W Application window texts – defects are identified by analysing texts that should be present in an application window. A single application window is analysed in isolation.
- A Application texts – defects are identified by analysing all application texts as a whole set. All application messages and texts are analysed.

Rules are small defect detection algorithms that are executed in a row during the main testing procedure. A testing procedure is presented in Algorithm 1.

Algorithm 1. Search for defects

Input: *source*, the application under test source code
Input: *app*, the compiled application under test
Input: *devices*, the smart phone devices list
Input: *rules*, defect detection rules
Output: *defects*, the list of defects, including device name, window, and defect type

```
defects ← Ø
paths ← getExecutionPaths(app)
foreach device ∈ devices do
  foreach path ∈ paths do
    executeTestPathOnDevice(device, path)
    foreach window ∈ path do
      textAreas ← recognizeTextAreas(window)
      recognizedText ← recognizeTexts(window, textAreas)
      actualTexts ← extractActualTexts(app, window)
      i18nTexts ← extractAllApplicationLocalizedTexts(app, window)
      textPlaceholders ← extractTextPlaceholders(app, window)
      foreach rule ϵ rules do
        defects ← defects ∪ checkRule(rule, textAreas, recognizedText, actualTexts, window, device, path, textPlaceholders)
```

When the main algorithm is executed, the *getExecutionPaths* procedure calculates all application navigation paths using application binary file. This action is performed using Droidbot tool and its algorithms [12]. This tool also executes generated testing paths on each selected testing device. During execution (*executeTestPathOnDevice* procedure), data for next steps are gathered by taking window screen-shots. When screenshots are taken, they are analysed in the following procedures:

1. *recognizeTextAreas* – identifies text locations on the screenshot, calculates text bounding boxes,
2. *recognizeTexts* – for each identified text area text recognition is performed [13] and text messages are extracted,
3. *extractActualTexts* – the application source code, resources, and runtime data are analysed, actual messages that should be displayed on a window are extracted,
4. *extractTextPlaceholders* – the application source code, resources, and runtime data are analysed, placeholders for localized actual messages that should be displayed on a window are extracted,
5. *extractAllApplicationLocalizedTexts* – analyses application files and extract all languages it supports, all localized, excluding current language, messages are grouped by languages.

When the screen-shot analysis is complete, each rule is executed against gathered analysis data. Each rule embeds small defect detection algorithm. Each rule uses different types of text sources and combinations of them. The list of rules and algorithms are presented in Table 3.

Here:

1. *calculateTextFontSize* – calculates texts font size in pixels,
2. *calculatePhysicalSize* – calculates physical text size on the screen of a selected device,
3. *spellCheck* – if text spelling is correct, returns true,
4. *grammarCheck* – if text grammar is correct, returns true,
5. *hasSynonym* – checks Thesaurus for synonyms of the word, checks if synonyms are present in all application texts,
6. *readabilityIndex* – calculates readability index using Gunning fog index [14] or any other [15] for the selected text,
7. *leadsSomeWhere* – returns if the link leads from the window to any other window within the given paths list,
8. *isStopWord* – checks if word is in the stop words list [16] and should be ignored,
9. isOffensive – checks if the text is offensive [17],
10. *isMisleading* – checks if the text is misleading [18],
11. *isVague* – trains neural network with bad messages [19], use to recognize similar messages,
12. *tooLongSentence* - checks if sentence has too many words, is it a compound sentence.

Table 3. Defect detection rules.

No	Defect Type	Rule source	Algorithms **Input:** *recognizedTexts*, recognized texts **Input:** *textAreas*, recognized texts areas **Input:** *actualTexts*, actual window texts in current language **Input:** *i18nTexts*, actual window texts in all app languages **Input:** *window*, analysed window **Input:** *device*, device screenshot was taken **Input:** path, execution path through app **Input:** *textPlaceHolders*, placeholders for translations **Output:** *defect*, detected defect
TP1	Text placement	S, W	**foreach** *area* ∈ *textAreas* **do** **if** *area* is **not** centred against surrounding controls **then** *defect* ← *window*, *device*, *area*
TS1	Font sizes	S, W	**foreach** *area* ∈ *textAreas* **do** *fontSize* ← *calculateTextFontSize(textArea)* **if** *fontSize* differ from surrounding controls text sizes **then** *defect* ← *window*, *device*, *area*

TS2	Unreadable text	S	**foreach** *text* ∈ *actualTexts* **do**
			if *recognizedTexts* **not** *contains text* **then**
			defect ← *window*, *device*, *text*
			foreach *area* ∈ *textAreas* **do**
			fontHeight ← *calculatePhysicalSize(textArea, device)*
			if *fontHeight* < *2mm* **then**
			defect ← *window*, *device*, *area*

TB1	Clashing background	S, W	**foreach** *text* ∈ *actualTexts* **do**
			if *recognizedTexts* **not** *contains text* **then**
			defect ← *window*, *device*, *area*
			if $\lvert text.color - window.backgroundColor \rvert < 5$ **then**
			defect ← *window*, *device*, *area*

TC1	Partial text	S, W	**foreach** *text* ∈ *actualTexts* **do**
			if *recognizedTexts* **not** *contains text* **then**
			foreach *recognizedText* ∈ *recognizedTexts* **do**
			if *text contains substring recognizedText* **then**
			defect ← *window*, *device*, *area*

TC2	Clipped text	S, W	**foreach** *text* ∈ *actualTexts* **do** **if** *recognizedTexts* **not** *contains text* **then** **foreach** *recognizedText* ∈ *recognizedTexts* **do** **if** *text contains substring recognizedText* **and** *recognizedText* endswith "…" **then** *defect* ← *window*, *device*, *area*
TE1	Wrong encoding	S, W	**foreach** *recognizedText* ∈ *recognizedTexts* **do** **if not** spellCheck(*recognizedText) text* **then** *defect* ← *window*, *device*, *area* **if** *recognizedText contains "?"* **and not** *end or start on "?"* **then** *defect* ← *window*, *device*, *area*
TM1	Missing text	S, W	**foreach** *text* ∈ *actualTexts* **do** **if** *recognizedTexts* **not** *contains text* **then** *defect* ← *window*, *device*, *area*
SD1	Synonyms usage	A, W	**foreach** *text* ∈ *actualTexts* **do** **foreach** *word* ∈ *text* **do** **if** *isNoun(word)* **and** *hasSynonym(word, actualTexts)* **then** *defect* ← *window*, *device*, *text*

SD3	Unclear terminology	A, W	*uniqueWords* ← extract distinct words from *actualTexts* **foreach** *word* ∈ *uniqueWords* **do** *translatedWord* ← *word* translated to another language *placeholders1* ← set of *textPlaceholders* where actualText contains word *placeholders2* ← set of *textPlaceholders* where i18nText contains *translatedWord* **if not** *placeholders1* matches *placeholders2* **then** *defect ← window, word, placeholders1, placeholders2*
SD4	Jargon	A, W	**foreach** *text* ∈ *actualTexts* **do** **foreach** *word* ∈ *text* **do** **if not** *isStopWord(word)* **and** *isNoun(word)* **and** *isTechnical(word, actualTexts)* **then** *defect ← window, device, text*
SS1	Bad text style	A, W	**foreach** *recognizedText* ∈ *recognizedTexts* **do** **if not** spellCheck(*recognizedText)* **or** **not** grammarCheck(*recognizedText)* **then** *defect ← window, device, recognizedText*
SS2	Too much text	W, A	**foreach** *text* ∈ *actualTexts* **do** *sentences* ← *text* split into sentences **foreach** *sentence* ∈ *sentences* **do** **if** *tooLongSentence*(*sentence)* **then** *defect ← window, sentence*
SS3	Vague error messages	W, A	**foreach** *text* ∈ *actualTexts* **do** *sentences* ← *text* split into sentences **foreach** *sentence* ∈ *sentences* **do** **if** *isVague*(*sentece)* **then** *defect ← window, sentence*
SS4	Untactful messages	W, A	**foreach** *recognizedText* ∈ *recognizedTexts* **do** **if** *isOffensive*(*recognizedText)* **then** *defect ← window, device, recognizedText*
SS5	Misleading text	W, S	**foreach** *recognizedText* ∈ *recognizedTexts* **do** **if** *isMisleading*(*recognizedText)* **then** *defect ← window, device, recognizedText*
SU1	Long instructions	W, S, A	**foreach** *area* ∈ *textAreas* **do** **if** *area* takes all window space **then** *defect ← window, device, area*
SU2	Too hard to understand text	A	**foreach** *text* ∈ *actualTexts* **do** **if** *readabilityIndex(text)* ≥ *CollegeLevel* **then** *defect ← text, window*

SU3	"…" misuse	W, A	**foreach** *text* ∈ *actualTexts* **do**
			if *text ends with "…"* **and** *text* is on *link* **then**
			if not *leadsSomeWhere(link, window, paths)* **then**
			defect ← text, window

SL1	Missing translations	S	**foreach** *recognizedText* ∈ *recognizedTexts* **do** **if** *textPlaceHolders contains recognizedText* **then** *defect ← window, device, recognizedText*

SL2	Wrong language	S	**foreach** *recognizedText* ∈ *recognizedTexts* **do**
			if *actualTexts* **not** *contains recognizedText* **then**
			if *actualTexts contains i18nTexts* **then**
			defect ← window, device, recognizedText

3.3 The Scheme for the Defect Detection

The main diving principle for detecting defects is to automate testing process. Here we propose a testing tool that executes the application under tests, gathers data, performs analysis and identifies possible defects. The main principle of the defect detection is structured as:

1. Generate tests that drive the application under test execution. The goal at this step is to generate tests that would cover 100% of the application under test code. There are several methods for generating tests [20–28] and tools [12, 29, 30].
2. Select a set of mobile devices to execute tests. The main selection criteria are to select as large as possible set of mobile devices with screens of varying physical sizes, resolutions, pixel densities, and colour depths. The devices could be emulated or real devices on a device cloud [31, 32].
3. Execute generated tests on each selected test device.
4. Gather screen-shot of each test step on each device.
5. Gather text messages using a live application window instrumentation of each test step on each device.
6. Run text recognition on each captured screen-shot and extract texts and their attributes:
   a. font size,
   b. position,
   c. size.
7. Extract application texts from its source code and corresponding application assets.
8. Run defects detection algorithms on each captured screen-shot.
9. Run defects detection algorithms on all gathered text messages from the application.

The scheme for defect detection describing tests generation, screenshots generation, and text analysis is presented in Fig. 1.

The main components of the scheme are:

1. DroidBot:
   a. Generates executable tests for navigating through the application under test.
   b. Extension to DroidBot that extracts application messages from the source code and corresponding assets.
2. Text executor:
   a. Runs all execution paths on all phones and emulators.
   b. Takes screen-shot at each path step on each device.
   c. Records device info, path step info, reference text.
3. Defect analyser:
   a. Analyses gathered screenshots and executes all defects detection algorithms.
   b. Provides a list of possible defects and their locations.

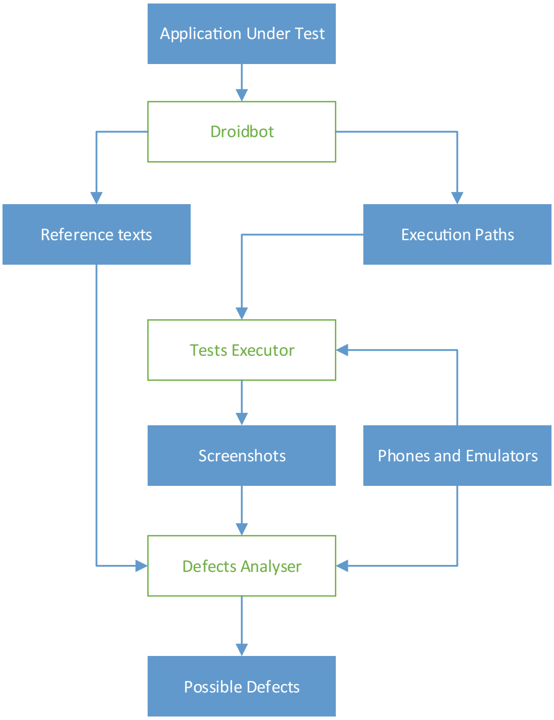


Fig. 1. User-interface defect detection scheme.

3.4 Testing Example

To present the proposed method of text defects several tools have been used. For test case generation the DroidBot tool has been applied.

For analysing captured screenshots custom tool has been developed: texts regions have been detected using the OpenCV [33] library by applying filters and afterwards the Tesseract [13] library has been used for extracting actual texts from screen-shots.

As a demo example the Android application with a multi-language interface has been selected. The Droidbot generated a set of execution paths and screenshots have been taken at every step. The created analysis tool has examined the captured screenshots and marked suspicious areas on them. Sample screen-shots are presented in Fig. 2.

The green bounds denote the detected text regions and the texts inside identify the discovered messages. Here, it is clear that the image A lacks several translations, while the image B has title and body messages in different languages (title is Norwegian, body is Czech languages).

The rules of text analysis were executed on each extracted message and its properties and defects were detected. The sample report is presented in Table 4. The tool detected a “too small text defect”. An internationalization problem was also detected. The message looked correct in the English language, but it was too long in Czech.

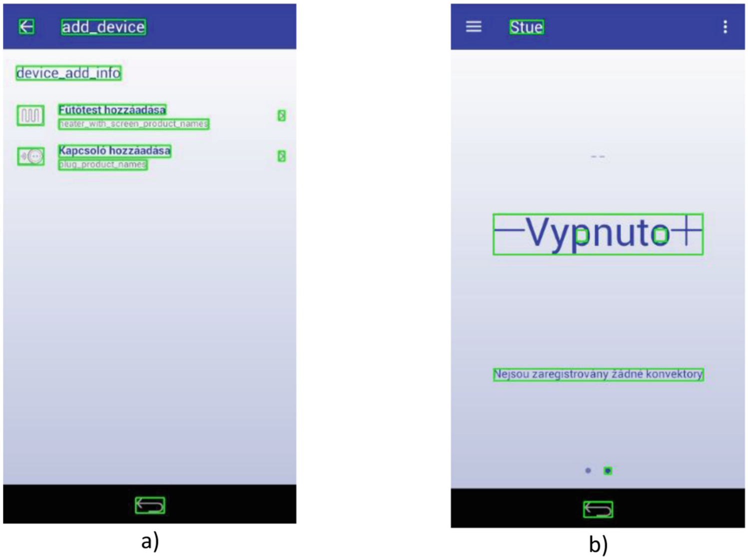
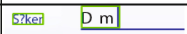
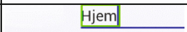


Fig. 2. Sample application defects.

Table 4. A report of the defects.

Defect	Device	Step/Window	Screenshot	Detected text	Expected Text
TS1	Jolla (EN)	Settings		Home name Hjem	Home name Hjem
TM1	Jolla (EN)	Settings		Hjem	Home name Hjem
TB1	Jolla (EN)	Settings		Home name Hjem	Home name Hjem
TC1	Xperia Ray	Settings		Hom name Hjem	Home name Hjem
TC2	Jolla (EN)	Settings		Home na… Hjem	Home name Hjem
TE1	Jolla (NO)	Settings		S?ker Hjem	Søker Hjem
TM1	Jolla (RU)	Settings		Hjem	Home name Hjem
SL1	Jolla (CZ)	Devices		Kapcsoló hozzáadása plug_product_na mes	Kapcsoló hozzáadása plug_product_na mes
TS2	Xperia Ray (2")	Settings		Home name Hjem	Home name Hjem
SL2	Jolla (CZ)	Control		Stue -Vypnuto-	Stue AV

Also, the text overlapped over other controls (+ and – buttons slightly overlap text and get fussed together).

The demo experiment was carried out using one application. It is planned to test on several hundred open-source applications [27] and compare findings with the lists of the already reported application issues.

4 Conclusions and Further Work

The method similar to an automated static analysis, and the tool framework for detecting user interface defects that mainly fall into text presentation and content defects categories is presented in the manuscript. The proposed method relies on image recognition, defect search and tested application execution on a big set of mobile devices. The main benefit of the application of the method is that it makes it possible to identify the troubling user interface locations, pinpoints the device type that could have the problem and provides visual proof of suspicion for further investigation.

The main drawback is long testing time – it is necessary to run all test cases on each device. One of the unanswered question is pointed at selecting an optimal subset of mobile devices for testing. Selection criteria could be a popularity of certain device configuration at a current of the future smartphones market.

The future work includes detailing and refining each defect detection rule and evaluating method effectiveness on a large set of mobile applications.

References

1. Packevičius, Š., Ušaniov, A., Stanskis, Š., Bareiša, E.: The testing method based on image analysis for automated detection of UI defects intended for mobile applications. In: Dregvaite, G., Damasevicius, R. (eds.) ICIST 2015. CCIS, vol. 538, pp. 560–576. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24770-0_48
2. Rasool, G., Arshad, Z.: A review of code smell mining techniques. J. Softw. Evol. Process **27**(11), 867–895 (2015)
3. Lelli, V., Blouin, A., Baudry, B.: Classifying and qualifying GUI defects. In: 2015 IEEE 8th International Conference on Software Testing, Verification and Validation (ICST) (2015)
4. Moran, K., et al.: Automated reporting of GUI design violations for mobile apps. In: 40th International Conference on Software Engineering (ICSE 2018) (2018)
5. Chang, T.-H., Yeh, T., Miller, R.C.: GUI testing using computer vision. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 1535–1544. ACM, Atlanta (2010)
6. Yeh, T., Chang, T.-H., Miller, R.C.: Sikuli: using GUI screenshots for search and automation. In: Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology, pp. 183–192. ACM, Victoria (2009)
7. Baek, Y.-M., Bae, D.-H.: Automated model-based Android GUI testing using multi-level GUI comparison criteria. In: Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering, pp. 238–249. ACM, Singapore (2016)
8. Chillarege, R., et al.: Orthogonal defect classification-a concept for in-process measurements. IEEE Trans. Softw. Eng. **18**(11), 943–956 (1992)

9. IEEE Standard Classification for Software Anomalies: IEEE Std 1044-2009 (Revision of IEEE Std 1044-1993), pp. 1–23 (2010)
10. Beizer, B.: Software Testing Techniques, 2nd edn. Van Nostrand Reinhold Electrical/Computer Science and Engineering Series. Van Nostrand Reinhold. New York (1990). xviii, 290 p.
11. Johnson, J.: GUI Bloopers 2.0: Common User Interface Design Don'ts and Dos. Morgan Kaufmann Publishers Inc., San Francisco (2007). 424 p.
12. Li, Y., et al.: DroidBot: a lightweight UI-guided test input generator for Android. In: Proceedings of the 39th International Conference on Software Engineering Companion, pp. 23–26. IEEE Press, Buenos Aires (2017)
13. Kay, A.: Tesseract: an open-source optical character recognition engine. Linux J. **2007**(159), 2 (2007)
14. Gunning, R.: The fog index after twenty years. J. Bus. Commun. **6**(2), 3–13 (1969)
15. Mailloux, S.L., et al.: How reliable is computerized assessment of readability? Comput. Nurs. **13**(5), 221–225 (1995)
16. Roman-Rangel, E., Marchand-Maillet, S.: Automatic removal of visual stop-words. In: Proceedings of the 22nd ACM International Conference on Multimedia, pp. 1145–1148. ACM, Orlando (2014)
17. Chen, Y., et al.: Detecting offensive language in social media to protect adolescent online safety. In: 2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Conference on Social Computing (2012)
18. Anderka, M., Stein, B., Lipka, N.: Detection of text quality flaws as a one-class classification problem. In: Proceedings of the 20th ACM International Conference on Information and Knowledge Management, pp. 2313–2316. ACM, Glasgow (2011)
19. Shneiderman, B.: Designing computer system messages. Commun. ACM **25**(9), 610–611 (1982)
20. Choi, W., Necula, G., Sen, K.: Guided GUI testing of android apps with minimal restart and approximate learning. SIGPLAN Not. **48**(10), 623–640 (2013)
21. Ganov, S.R., et al.: Test generation for graphical user interfaces based on symbolic execution. In Proceedings of the 3rd International Workshop on Automation of Software Test, pp. 33–40. ACM, Leipzig (2008)
22. Machiry, A., Tahiliani, R., Naik, M.: Dynodroid: an input generation system for Android apps. In: Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering, pp. 224–234. ACM, Saint Petersburg (2013)
23. Amalfitano, D., et al.: Using GUI ripping for automated testing of Android applications. In: Proceedings of the 27th IEEE/ACM International Conference on Automated Software Engineering. ACM (2012)
24. Memon, A., Nagarajan, A., Xie, Q.: Automating regression testing for evolving GUI software. J. Softw. Maintenance Evol. Res. Pract. **17**(1), 27–64 (2005)
25. Memon, A., Banerjee, I., Nagarajan, A.: GUI ripping: reverse engineering of graphical user interfaces for testing. In: 2013 20th Working Conference on Reverse Engineering (WCRE). IEEE Computer Society (2003)
26. Xun, Y., Memon, A.M.: Generating event sequence-based test cases using GUI runtime state feedback. IEEE Trans. Softw. Eng. **36**(1), 81–95 (2010)
27. Su, T., et al.: Guided, stochastic model-based GUI testing of Android apps. In: Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering, pp. 245–256. ACM, Paderborn (2017)

28. Miguel, J.L.S., Takada, S.: Generating test cases for Android applications through GUI modeling, usage modeling, and change analysis. In: Proceedings of the Eighth International C* Conference on Computer Science and Software Engineering, pp. 146–147. ACM, Yokohama (2008)
29. Su, T.: FSMdroid: guided GUI testing of android apps. In: Proceedings of the 38th International Conference on Software Engineering Companion, pp. 689–691. ACM, Austin (2016)
30. Zeng, X., et al.: Automated test input generation for Android: are we really there yet in an industrial case? In: Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering, pp. 987–992. ACM, Seattle (2016)
31. Baride, S., Dutta, K.: A cloud based software testing paradigm for mobile applications. SIGSOFT Softw. Eng. Notes **36**(3), 1–4 (2011)
32. Starov, O., Vilkomir, S., Kharchenko, V.: Cloud testing for mobile software systems-concept and prototyping. In: ICSOFT (2013)
33. Chiatti, A., et al.: Text extraction from smartphone screenshots to archive in situ media behavior. In: Proceedings of the Knowledge Capture Conference, pp. 1–4. ACM, Austin (2017)

The Impact of the Cost Function on the Operation of the Intelligent Agent in 2D Games

Dawid Połap(✉) and Marcin Woźniak(✉)

Institute of Mathematics, Silesian University of Technology,
Kaszubska 23, 44-100 Gliwice, Poland
{Dawid.Polap,Marcin.Wozniak}@polsl.pl

Abstract. A large part of the technology development depends on the needs of users. Apart from the hardware requirements for programs used by large companies or smaller groups, the wide applications and hardware load are games and graphics. Increasing the quality of games by improving their story quality requires a lot of more efficient and effective algorithms. In this work, we propose the use of a hybrid approach to the management of opponents' movements on the classic two-dimensional game called the Tron. Our solution is based on the use of the idea of a simulated annealing algorithm in order to select the agent's movement technique depending on the cost function. The algorithm has been implemented and tested depending on the used parameters. Obtained results were discussed depending on the advantages and disadvantages of using this type of solution in more complex games.

Keywords: Cost function · Tron game · Arcade game · Intelligence agent

1 Introduction

Games are considered to be a motivator not only for computer hardware manufacturers, what can be seen from the competition of companies producing graphics cards such as NVIDIA or AMD. A lot of emphasis is also placed on the creators of stories and activities of specific elements that affect the gameplay. Even the most complex games can use the same solutions as simple games. For the purposes of solution design and testing, the simplest games work best. An example are a simple arcade games that analyzing a player's movement that can adjust the level of an opponent controlled by the algorithm. Examples of games that are popular among programmers are primarily Pacman or Tron.

Creating dedicated solutions for increased playability can be considered in several different categories. Starting with the selection of hardware, programming language [1] or techniques. Particularly the technique is an important element, because the artificial intelligence algorithms find more and more applications in the industry. The depiction of this process is visible on the amount of work under various grants. Scientists find use in medicine, for instance in [2] where heart activity is analyzed. The process of training artificial intelligence requires a lot of other activities. Input data should be processed in

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 293–302, 2018.
https://doi.org/10.1007/978-3-319-99972-2_23

such a way that it would be possible to classify them in a simple way. In [3], the problem of bathymetric big data interpolation was presented. The application of such a problem was considered as a tool used in navigation. On the other hand, all data must be kept somewhere. For this purpose, databases are used that guarantee quick access to them. This is important for learning systems where data has to be divided into two parts – teaching and testing. Research on data storage and fast access are shown in [4, 5]. The development of techniques based on the classic game is shown in [6]. Again in [7], the authors shown analysis for price-based demand response, and in [8], general game rule generation problem is described with a proposition of framework for that. Important topic is the pursuit after the player in different games, which was investigated in terms of communication between several agents in [9]. Research on agents was also presented in [10]. An interesting approach is to learn the algorithm using screenshots from the game by using convulsive neural networks [11]. The strategies used in games are reflected in gamification, where players' behavior can be modeled according to the technique adopted [12].

In this paper, we present a proposal of an agent's activity in games that simulates the action of the opponent. The proposition is based on the use of the idea of simulated annealing and cost function depending on the selected parameters.

2 Intelligent Agent Operation Based on Cost Function

The agent will be called the algorithm responsible for the operation of the opponent in the game with the player. The game's environment is a board on which the player can make some movement. The implemented agent can be rational if the value of the quality measure is maximized, assuming that it is not omniscient. We assume that the agent uses only the knowledge available on the observation basis. This quality measure can be understood as some security or profit for the agent. Moreover, an agent can make decisions in an episodic or sequential manner. In the case of the first one, the episode is understood as the observation in a given position which is a consequence of the previously made environmental analysis. In the case of sequencing, the decision is based in large measure on the previous action, and the current one will affect the next move.

2.1 Tron Game

Tron is the modification of the classical snake game. It is inspired by the film with the same title from 1982 directed by Steven Lisberger. The game is a top view for some kind of motor racing. The idea is that each user moves with one motor and leaves his trail behind. The player who stays the last on the board wins. The condition is that the other players will collide at some point with any of the traces left behind.

2.2 Proposition of Intelligent Agent's Operation

Continuous following the player is one of the basic methods of moving in this type of games. However, it should be noted that such an action can quickly get bored of the playing user. To remedy this, the agent's activity should be considered from several different strategies depending on their momentary quality. Knowledge about the board allows to calculate the quality of the next move due to a certain function. Let's assume that we have a set of certain motion strategies. Their selection cannot be made in every movement because of the left trace. If it would be allowed, the movement would look skewed, and the basic principle is perpendicular motion. Let's choose the strategy depending on the number of moves α. If the rest of division alpha by a particular value, for example *80* is equal to zero, then the selection is made to choose one of the strategy. Each strategy returns a possible direction of movement, which is evaluated by the following function

$$f\left(\bar{x}_{target}, \bar{x}_{poss}\right) = \epsilon d\left(\bar{x}_{target}, \bar{x}_{poss}\right) + \vartheta g\left(\bar{x}_{target}, \bar{x}_{poss}\right), \tag{1}$$

where $\bar{x}_{target}$ is the target point, and $\bar{x}_{poss}$ is a possible position indicated by the algorithm, coefficients ϵ and ϑ are weights of a given function that satisfy the equation $\epsilon + \vartheta = 1$. The function $d(\cdot)$ is one of the metric distance between a given, two points, and $g(\cdot)$ is the numerator of fields on which collision may occur during the designated trace to the destination point.

The above assessment function allows to evaluate the direction chosen by the given strategy. The obvious assumption is that there are at least two strategies. In the case of one, the proposed solution is unnecessary and only increases the number of calculations. In the case of a larger amount, the strategy is selected and continued for several dozen moves. *2* or *3* strategies are optimal for this type of game, because it allows to vary the agent's action, as well as reduce the likelihood of predicting his movements by the player. Our proposal is to adopt three strategies - after choosing one of them, the movement takes place through the above-mentioned *80* steps. Each strategies is evaluated and the one whose value is the smallest is chosen.

Algorithm 1. Agent's operation in the Tron game

```
1: Start,
2: Define cost function f(·),
3: Define actual direction of movement,
4: Calculate the cost of pursuit after the player as ξ1,
5: Calculate the cost of simulated annealing strategy as ξ2,
6: Calculate the cost of random movement as ξ3,
7: Find the lowest cost from the set {ξ1, ξ2, ξ3},
8: Choose the direction with the lowest cost,
9: Stop.
```

Three strategies that can be applied to the agent's operation are pursuit, random movement and one based on simulated annealing strategy.

Pursuit strategy causes that the agent follows the shortest path to the player. Considering the fact that the pursuit of the exact point of the user is pointless - because it can end with collision due to the fault of the agent and mean his loss. We suggest pursuit towards a point in a certain neighborhood of the player. For the neighborhood equal to *3*, the grid will have a 3×3 dimension where the middle point is the player's position. One of nine points is randomly selected and labeled as $\bar{x}_{target}$. Then all possible positions in the agent's environment are checked. If the distance relative to a given metric is smaller, then this direction is chosen. The full algorithm is presented in Algorithm 2.

Algorithm 2. Pursuit strategy

Start,
Define actual position as $\overline{x}_{act}$,
Find a random neighbor from the opponent's surroundings $\overline{x}_{target}$,
Define new variable with actual movement as $\overline{x}_{dir}$
Define actual direction as dir,
for each possible movement $\overline{x}_{poss}$ **do**
    Calculate the distance between $\overline{x}_{target}$ and $\overline{x}_{poss}$ using metric function $d(\cdot)$,
    if $d(\overline{x}_{target}, \overline{x}_{poss}) < d(\overline{x}_{target}, \overline{x}_{dir})$ **then**
        $\overline{x}_{dir} = \overline{x}_{poss}$,
        Change possible direction in dir,
    end if
end for
Move according to the found direction dir,
Stop.

Next strategy is a random movement. It the simplest way to change the direction, it is can be used in two situation. As a strategy and in the situation, when the agent is against some obstacle like a leaved trace. The mechanism is simply – the agent can move in four basic direction where one of them was the current and the opposite was his route. And so there are two other directions where it can go (both at 90° angle to the current one). The direction is chosen randomly. Preventing an impact on a possible wall for the selected position may cause a lack of movement, and thus encounter its own trace. The algorithm of these strategy is shown in Algorithm 3.

The third strategy borrows the main idea from simulated annealing algorithm [13], which is based on a metallurgical process involving the heat treatment of metal. Physically, this involves heating the metal and then cooling it to bring it closer to the level of equilibrium. The use of an algorithm for an agent described in this work, i.e. having limited movements, forces some simplification of it. Let's assume that algorithm have the position of the player $\bar{x}_{player}$, the agent $\bar{x}_{act}$ and the initial temperature T. Similarly to the pursuit strategy, the strategy will search a certain point in a grid of size 3×3, where the player is in the center. For each neighbor and each possible movement, the distance is calculated. If the value μ_{new} is smallest, it is replaced and the direction is changed. But there is possibility, that from a random value x in the range $<0,1>$ will be satisfy the following equation

$$x < \exp\left(\frac{\mu_{new}}{T}\right), \tag{2}$$

what is understood as accepting the solution with a certain probability. This makes it possible to lose by an agent who is not infallible. After each checked point, the temperature value is reduced using

$$T = 0.98 \cdot T. \tag{3}$$

Algorithm 3. Random movement strategy

Start,
Define actual direction as dir,
if the next move means a collision **then**
  Choose one of the possible directions at random,
end if
Stop.

Algorithm 4. Simulated annealing strategy

Start,
Get actual position $\overline{x}_{act}$,
Define a start temperature T,
Define a size of neighborhood $a \times a$ where the target point is in the center,
Define new variable with actual movement as $\overline{x}_{dir}$ and direction dir,
Calculate distance $d(\cdot)$ between $\overline{x}_{poss}$ and target point as μ,
for each neighbor **do**
  for each possible move $\overline{x}_{poss}$ **do**
    Calculate distance μ_{new} between $\overline{x}_{poss}$ and the neighbor using one of a given metric,
    if $\mu > \mu_{new}$ **then**
      if possible move is not a crash **then**
        $\mu = \mu_{new}$,
        Change the direction in dir,
      end if
      Choose random value x in the range $\langle 0, 1 \rangle$,
    else if $x < \exp\left(\frac{\mu_{new}}{T}\right)$ **then**
      $\mu = \mu_{new}$,
      Change the direction in dir,
    end if
  end for
end for
Decrease the temperature T using Eq. (3),
Stop.

3 Experiments

The proposed mechanism was tested due to different parameters, particular attention was paid to the selection of metrics in various configurations. For experiments purposes, described solution was tested for each set of parameters 100 times (until one player lost). To enable obtaining the most effective testing techniques, test were made under two types of action – with the player (50 tests) and automatic (50 tests). As automatic tests, the game of two players controlled by two separated algorithms is understood. Selected game results are presented in Fig. 3.

In Tables 1 and 2, obtained time measurements are presented. For each algorithm, 25 test were performed, where selected metric was Euclidean or Manhattan. For a given two points $\bar{x}_a = (x_a, y_a)$ and $\bar{x}_b = (x_b, y_b)$, Euclidean metric is defined as

$$d_E(\bar{x}_a, \bar{x}_b) = \sqrt{(x_a - y_a)^2 + (x_b - y_b)^2}, \quad (4)$$

and Manhattan metric as

$$d_M(\bar{x}_a, \bar{x}_b) = |x_a - y_a| + |x_b - y_b|. \quad (5)$$

Table 1. Average time comparison depending on the operation technique with Euclidean metric.

		Algorithm			
		Pursuit strategy	Random movement strategy	Simulated annealing strategy	Agent's operation described in Algorithm 1
Average time [s]	Algorithm vs human player	13	7	9	28
	Algorithm vs algorithm	11	6,5	9,5	35

Results shows that it is better to use Manhattan metric which extend the playing time. Especially when agent can change the strategy. The Euclidean metric was better when using a specific strategy, and not with an extended algorithm. However, the playability is smaller, and the predictability of the opponent's movements is much higher. In the case of applying random movement, the agent was looped quite often causing a collision with its own trace – in both cases (Table 3).

The number of player/used algorithm wins was also counted depending on the selected metric. The results are shown in Fig. 2 and 3. The division of wins and losses

Table 2. Average time comparison depending on the operation technique with Manhattan metric.

		Algorithm			
		Pursuit strategy	Random movement strategy	Simulated annealing strategy	Agent's operation described in Algorithm 1
Average time [s]	Algorithm vs human player	8	3	8	37
	Algorithm vs algorithm	13	2	11	47

Table 3. The probability of player's winning.

	Algorithm			
	Pursuit strategy	Random movement strategy	Simulated annealing strategy	Agent's operation described in Algorithm 1
Euclidean	60%	96%	52%	44%
Manhattan	56%	92%	44%	36%

is distributed almost the same for individual algorithms, i.e. the choice of a specific measure of distance does not affect this aspect too much. In the case of random movement, the number of player's wins is not a surprise, because the algorithm loses in the first few moves. The significant advantage of winning the algorithm is visible especially for the intelligent agent, where in both metrics there is an advantage over the player. For the Manhattan metric, the average advantage of winning the algorithm in relation to the user is almost *1.8* (Fig. 1).

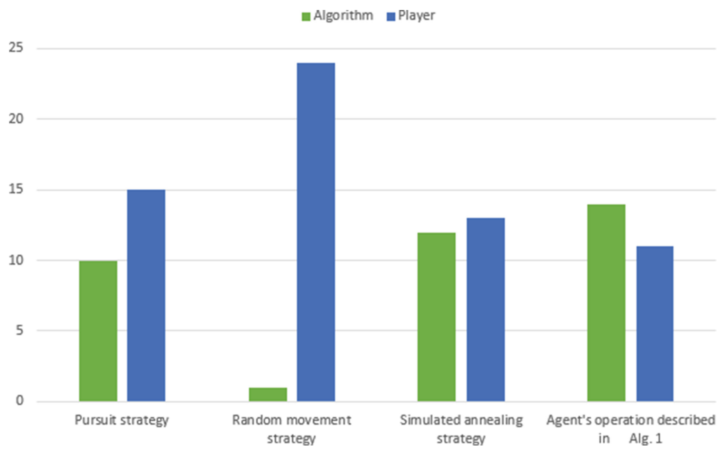


Fig. 1. The number of wins depending on the used algorithm with the Euclidean metrics.

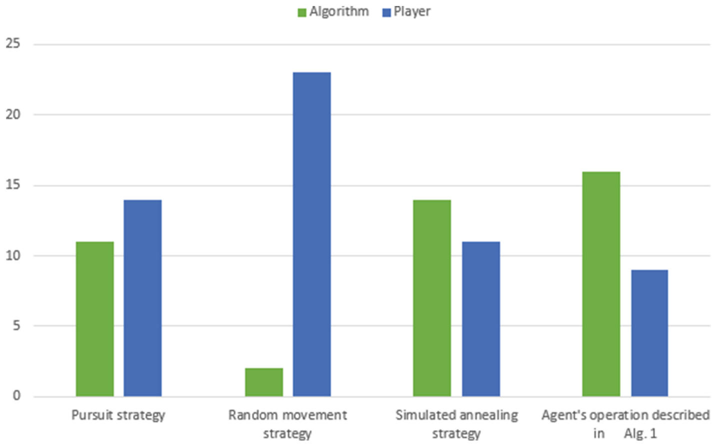


Fig. 2. The number of wins depending on the used algorithm with the Manhattan metrics.

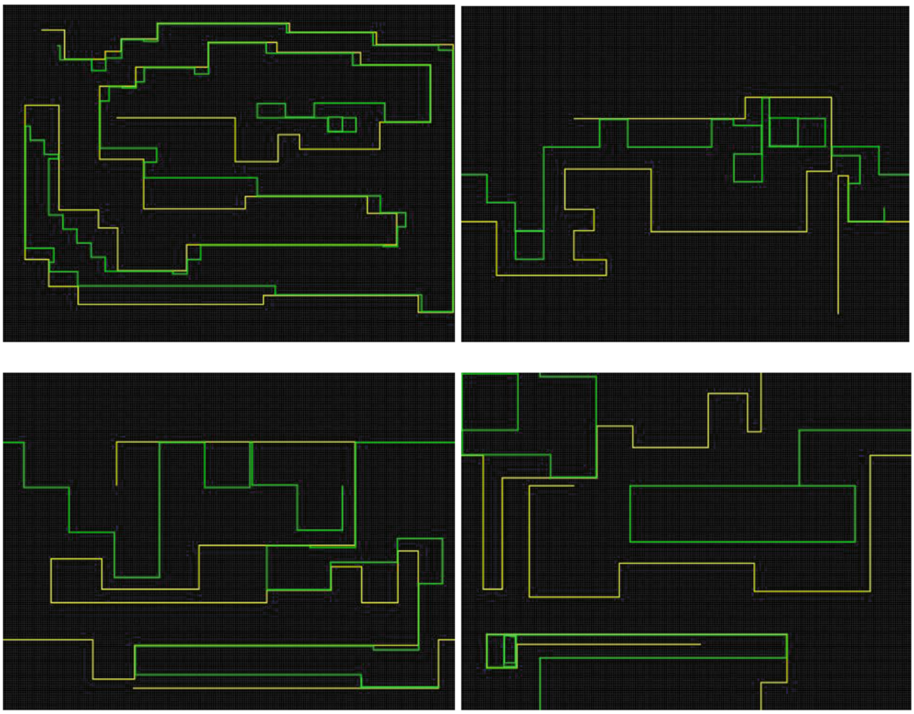


Fig. 3. Sample boards while playing with the proposed agent.

4 Conclusion

Algorithms and techniques of intelligent agents in the application of the simplest games give great opportunities for wider use. It is important that the techniques are universal. From small, simple games, algorithms can be easily transferred to large-scale entertainment. In addition, all activities in this area, give the opportunity to build other systems of behavior analysis and use within the scope of gamification. In this paper, we presented the idea of agent's operation on the basis of minimizing the value of the cost function, which assessed the direction of movement determined by the given strategies. The proposed technique uses the selected metric, here we tested the action within two of them – Euclidean and Manhattan. The obtained results showed that the proposal regarding to any metric is valuable (although it can be seen that the Manhattan metric is better one which may be the caused by the perpendicular movements of the players). Not only the level of gameplay increases due to the unpredictability of the agent's moves, but extends the time of the game. And this in turn causes that the player must focus more on the game if he does not want to lose.

Acknowledgments. Authors acknowledge contribution to this project of the "Diamond Grant 2016" No. 0080/DIA/2016/45 from the Polish Ministry of Science and Higher Education and the Rector pro-quality grant No. 09/010/RGJ18/0033 at the Silesian University of Technology, Poland.

References

1. Chodarev, S., Bacíková, M.: Development of Oberon-0 using YAJCo. In: 2017 IEEE 14th International Scientific Conference on Informatics, pp. 122–127. IEEE (2017)
2. Beritelli, F., Capizzi, G., Sciuto, G.L., Napoli, C., Scaglione, F.: Automatic heart activity diagnosis based on gram polynomials and probabilistic neural networks. Biomed. Eng. Lett. **8**(1), 77–85 (2018)
3. Włodarczyk-Sielicka, M., Wawrzyniak, N.: Problem of bathymetric big data interpolation for inland mobile navigation system. In: Damaševičius, R., Mikašytė, V. (eds.) ICIST 2017. CCIS, vol. 756, pp. 611–621. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67642-5_51
4. Marszałek, Z.: Parallelization of modified merge sort algorithm. Symmetry **9**(9), 176 (2017)
5. Gabryel, M.: A bag-of-features algorithm for applications using a NoSQL database. In: Dregvaite, G., Damasevicius, R. (eds.) ICIST 2016. CCIS, vol. 639, pp. 332–343. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46254-7_26
6. Rohlfshagen, P., Liu, J., Perez-Liebana, D., Lucas, S.M.: Pac-Man conquers academia: two decades of research using a classic arcade game. IEEE Trans. Games (2017)
7. Ye, M., Hu, G.: Game design and analysis for price-based demand response: an aggregate game approach. IEEE Trans. Cybern. **47**(3), 720–730 (2017)
8. Khalifa, A., Green, M.C., Perez-Liebana, D., Togelius, J.: General video game rule generation. In: 2017 IEEE Conference on Computational Intelligence and Games (CIG), pp. 170–177. IEEE (2017)
9. Grossi, G., Ross, B.: Evolved communication strategies and emergent behaviour of multi-agents in pursuit domains. In: 2017 IEEE Conference on Computational Intelligence and Games (CIG), pp. 110–117. IEEE (2017)

10. Esmaeili, A., Mozayani, N., Motlagh, M.R.J., Matson, E.T.: A socially-based distributed self-organizing algorithm for holonic multiagent systems: case study in a task environment. Cogn. Syst. Res. **43**, 21–44 (2017)
11. Kunanusont, K., Lucas, S.M., Pérez-Liébana, D.: General video game AI: learning from screen capture. In: 2017 IEEE Congress on Evolutionary Computation (CEC), pp. 2078–2085. IEEE (2017)
12. Ašeriškis, D., Damaševicius, R.: Gamification patterns for gamification applications. Procedia Comput. Sci. **39**, 83–90 (2014)
13. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. Science **220**(4598), 671–680 (1983)

The Research on Method of Prediction Mine Earthquake Based on the Information Entropy Principle

Baoxin Jia[1,2(✉)], Linli Zhou[1], Yishan Pan[3], Chengfang Yang[1], and Zhiyong Li[1]

[1] School of Civil Engineering, Liaoning Technical University, Fuxin 123000, China
13941855200@126.com
[2] Institute of Geology, China Earthquake Administration, Beijing 100000, China
[3] Liaoning University, Shenyang 110136, China

Abstract. Earthquake prediction is researched by using the information entropy principle, which provides that magnitude distribution model is not in conformity with G-R or fractal index model, and the reason that the mine earthquake magnitudes obey a certain probability distribution is explained. It is presented to calculate the corresponding information entropy taking advantage of existing mine earthquake measuring results, therefore, the occurrence of mine earthquake is forecasted according to calculation result of entropy. The mine earthquake takes place easily when entropy reduces. Forecast method is tested by the monitoring data of mine earthquake, and the result shows that the method is feasible. Our results provide a kind of effective method of mine earthquake statistical distribution modeling and information entropy prediction of mine earthquake.

Keywords: Information entropy · Probability distribution · Mine earthquake Prediction model

1 Introduction

Mine earthquake is induced by mining, classified into mine dynamic disasters, which was a phenomenon of elastic deformation energy release during the mining process companying loud noise and shock wave vibration. The destructive vibration of the mining area caused by rock instability, such as impact ground pressure, coal and gas outburst process, and large area roof caving are generally referred to as the mine earthquake. When mine earthquake occurs, the rapid release of energy of surrounding rock, sudden breaking of coal and rock lead to roof caving, support breaking, roadway congestion, ground vibration, building damage and human casualty. At present mines in China are mostly built in 50–60 s of the 20th century which are about to enter deep mining so mine earthquake risk will become more and more serious. It is particularly important for mine earthquake prediction to conduct feasible and effective analysis according to the result of mine earthquake monitoring [1–4].

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 303–311, 2018.
https://doi.org/10.1007/978-3-319-99972-2_24

Mine earthquake disaster system is a very complicated system. The occurrence and evolution of have rather complex features, such as ordering, kick, irreversibility, unpredictable for a long time, and fuzzy and grey characteristics. Generation model, influence factors, the precursor information and distribution model have the nature of uncertainty. It is no very clear for the reason of mine earthquake and the influence factors. It is also difficult problem to forecast the time, location and size of mine earthquakes occurrence. In terms of mine earthquake, before it occurs, prediction can be made by features that b value of G-R model of mine earthquake reduces distributed according to time and space of mine earthquake and reduction of fractal dimension of fractal model. Although these methods can roughly identify the importance of the various factors, actual time and space distribution of mine earthquake activity is not completely accords with G-R model and the fractal model. The fractal structure in the natural world is different from ideal fractal model whose self-similarity can only be founded in a limited range of the size and statistical significance. The same is true of the self-similarity of fractal structure in the mine earthquake. The b value of GR in the mine earthquake is founded only within the scope of the size and statistical sense. Upper and lower size limit, especially the lower bounds, has a great influence on the results of the analysis so it is difficult to accurately predict the degree of various influencing factors toward the law of mine earthquake. Prediction error is relatively bigger [5–9]. Aiming at the existing problem of mine earthquake prediction methods, considering the uncertainty of mine earthquake events, this paper proposed the prediction method of mine earthquake occurred in the process of coal mining based on the entropy theory, in order to achieve the purpose of forecasting the mine earthquake.

2 Entropy and Information Entropy

Entropy, a measure of uncertainty, is one of the important concepts of physics. In 1896, the Boltzmann and Plank relate entropy and system on the number of micro state to illustrate the statistical significance of entropy which makes the second law of thermodynamics be shown in the form of an isolated system entropy increase principle. Since then the entropy as a criterion of the judgment all direction and limit of thermodynamic process is widely used in natural science [11, 12]. In 1948, Shannon applied the entropy concept in thermal statistics to the field of information to indicate the uncertainty of information source. The information entropy is derived from thermodynamic entropy (physical entropy), and it is a kind of the application and development of information science. The information entropy is a concept used to measure the amount of information. The more orderly a system is, the less the information entropy is; conversely, the more chaos a system is, the higher the information entropy is. So the information entropy is also considered as the measurement of ordering degree of a system. If used to show probability of some signal in an information source, the information entropy is defined as:

$$H(x) = -\sum_{i=1}^{n} p(x_i) \ln p(x_i) \tag{1}$$

The information entropy here characterizes the features of the information source, and it is a single value of distribution probability function with the additive property which can be used as information measure of uncertainty [13–15].

3 Information Entropy and Distribution of Mine Earthquake Magnitude

Anything is associated with a certain amount of time and space. There is nothing out of time and space. In the characteristics studies of preparation, occurrence of mine earthquake and rock burst, the distribution is characterized by order with formation of crack and fracture according to change of stress field. This is characteristics demonstrated in the process which is from disorder to order, and eventually lead to mutation in an opening interactive system [8–10]. Prior to the mine earthquake, this kind of orderly change is characterized by the decrease of information entropy in the time and space distribution of mine earthquake. As long as we define information entropy of the mine earthquake, we can use it to predict the occurrence of mine earthquake by discussing orderly or disorderly development of evolution system.

For mine earthquake information entropy of discrete random variables, the information entropy of mine earthquake magnitude is set as the corresponding probability of mine earthquake occurrence:

$$S(x) = -\sum_{i=1}^{n} p(x_i) \ln p(x_i) \tag{2}$$

For the information entropy of continuous random variables:

$$S(x) = -\int_{-\infty}^{+\infty} p(x) \ln p(x) dx \tag{3}$$

Probability distribution function $p(x)$ Satisfies the constraint conditions:

$$p(x) \geq 0 \tag{4}$$

$$\int p(x) dx = 1 \tag{5}$$

Format (3) represents the meaning of the following two aspects: if the probability distribution function of information occurrence is available, the entropy can be calculated according to the format, which can be taken as a function of probability distribution. When change happens, change accordingly; If information entropy has been known, we can calculate the probability distribution function.

If mathematical expectation for x is, standard deviation is, demand density function should satisfy in addition to satisfy constraints (4), (5):

$$\int xp(x)dx = \mu \tag{6}$$

$$\int x^2 p(x)dx = \sigma^2 \tag{7}$$

Using Lagrange undetermined multiplier method to set up the Lagrange function equation, the corresponding maximum probability distribution of entropy is available. By this way, we can prove

$$S(x) = -\int \left[p(x)\ln p(x) + \alpha p(x) + \beta x^2 p(x)\right]dx \tag{8}$$

Make the change be zero, take the maximum information entropy

$$\frac{\partial S(x)}{\partial p(x)} = -1 - \ln p(x) + \alpha + \beta x^2 = 0 \tag{9}$$

Selecting constants α and β to satisfy the constraint condition, the corresponding maximum entropy probability distribution is

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{10}$$

The above type is normal distribution. Relationship between entropy and standard deviation for normal distribution is

$$S(x) = \ln(\sqrt{2\pi}e \cdot \sigma) \tag{11}$$

If the mathematical expectation of x is, density function is required to satisfy 3 constraints of (4), (5), (6), corresponding maximum entropy probability distribution can be inferred:

$$p(x) = \frac{1}{\mu} e^{-\frac{x}{\mu}} \tag{12}$$

The above type is exponential distribution. Relationship between entropy and standard deviation for exponential distribution is

$$S(x) = \ln(1 \cdot \sigma) \tag{13}$$

Visible information entropy model contains the G-R and fractal index model. Other uniform distribution, Weibull distribution and logarithmic distribution can also be launched by information entropy theory [16–19], and the relation between entropy and the standard deviation also meets the same form, which can be shown as

$$S(x) = \ln(k \cdot \sigma) \tag{14}$$

Just different distribution corresponds to different k value. By entropy and standard deviation of the actual test data, we can calculate the k values and determine distributing type. It also illustrates the relation between entropy, the distribution form and the standard deviation—not only each sub-state but also distribution changes according to the state change of open system. Entropy can reflect comprehensively changes of system state. It is kind of unified and feasible prediction model on the basis of information entropy which is not restricted by distribution model.

In 2016, we took nearly one year to monitor and locate mine earthquake occurred in mining area 1305 of Dongtan Coal Mine, recording time, location and size of earthquakes. Using the SOS microseismic monitoring system, the system can adapt to the complex working environment in the mine, and has independent underground power supply mode. The underground equipment is simple, which can avoid excessive underground equipment from generating high equipment failure rate. In addition, the system uses an independent, unified GPS timing mode to achieve real-time and synchronization of all vibration sensors in the underground. Figure 1 is the relation curve about probability density distribution and magnitude of the mine earthquake which is derived from measuring results of more than 4000 mine earthquakes. We use magnitude 0.1 to divide the magnitude level. As the chart shows, the relation between occurrence and magnitude of mine earthquakes does not conform to exponential distribution (G-R or fractal index model), only when magnitude is bigger than 0.5 does it approximately conform, and does not conform well to the normal distribution. A large amount of useful information will be lost if data bigger than level 0.5 are used to make statistical analysis on G-R or fractal. Because mine earthquake, exceeding 0.5 on the Richter scale poses a hazard to safety of mines, we must use mine earthquake data under magnitude 0.5 to predict mine earthquake exceeding 0.5 on the Richter scale. The space-time distribution of mine earthquake is in the form of the changes, so the statistical analysis on of the fixed distribution is unreasonable. In view of above discussion on the rationality and uniformity of the forecast based on the information entropy, we put forward the application of entropy theory to the prediction of mine earthquake in order to take effective measures in advance to prevent the disasters.

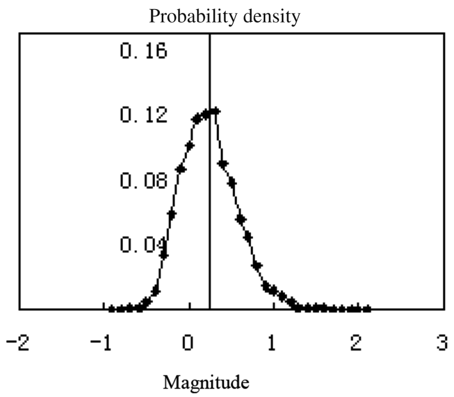


Fig. 1. The probability distribution of mine earthquake magnitude

4 Application of Entropy Theory in Mine Earthquake Prediction

According to the entropy theory, in the micro sense, entropy is a measure degree of disorder and chaos. The more unbalanced and orderly system is, the smaller the entropy is, and vice versa. The equilibrium entropy is the biggest of all. For mine earthquake, system after mine earthquake is the most chaotic, which is called equilibrium. The entropy should be the biggest at this stage. I.e. when entropy increases, the danger has passed while entropy reduces. It is a prelude to danger. So we can use the decrease of entropy as the basis of mine earthquake prediction in advance [20, 21].

As for entropy's calculation, we choose certain dense areas near the working face (working face length 200 m, prior 50 m posterior 50 m), taking half a month for a computational time sliding section. In order to remove influence on earthquake information entropy because of difference between the different time system states, we adopt separated and relative information entropy of mine earthquake when calculate, namely

$$S = S(x)/\ln n = \frac{-\sum_{i=1}^{n} p(x_i)\ln p(x_i)}{\ln n} \tag{15}$$

N stands for total magnitude levels. When probability of each magnitude is equal, probability is $1/n$, $\ln n$ is the maximum entropy.

We have collected the cumulative number of mine earthquakes and the highest energy level of mine in the nearly one year in 2016, as shown in Table 1. It is not difficult to find from the table that in the 10th–13th week, the 21st–24th week, and the 34th–37th week, the number of mine earthquakes is obviously higher than other time periods, and the mine earthquake energy level reaches up to 10^7 J. In contrast to the mining work, we found that within the above three time periods, the mining activities are located in faults that are prone to mine earthquakes.

Table 1. Cumulative number of mine earthquakes and the highest energy level statistics for 1305 working face in Dongtan Coal Mine in 2016

Record time (week)	Cumulative number of mine earthquake	The highest energy level of mine earthquake (J)	Record time (week)	Cumulative number of mine earthquake	The highest energy level of mine earthquake (J)
1	17	10^3	23	82	10^7
2	14	10^3	24	65	10^7
3	13	10^3	25	54	10^6
4	15	10^3	26	32	10^4
5	12	10^3	27	53	10^4
6	10	10^4	28	62	10^6
7	20	10^5	29	58	10^5
8	25	10^5	30	28	10^3
9	42	10^6	31	21	10^4
10	55	10^7	32	32	10^4
11	63	10^7	33	41	10^5
12	76	10^7	34	56	10^6
13	73	10^7	35	76	10^7
14	56	10^6	36	79	10^7
15	47	10^6	37	59	10^6
16	21	10^4	38	47	10^5
17	32	10^5	39	49	10^6
18	26	10^4	40	55	10^6
19	32	10^5	41	58	10^6
20	57	10^6	42	43	10^6
21	68	10^7	43	39	10^4
22	76	10^7	44	32	10^4

After that, we calculated the mine earthquake information entropy based on the monitoring data, Fig. 2 is mine earthquake information entropy on the base of measured data. From the figure, the information entropy is different in different measuring time. There are three obvious decreases of entropy, respectively corresponding to the period of 75 days to 90 days, the period of 150 days to 165 days and 240 days to 255 days. The time period during which these three entropy decreases is consistent with the time period during which the mine earthquake occurs more frequently. In these three phases, because the mining activity is in the location of faults prone to mine earthquake, the number of mine earthquakes increases, the mine earthquake energy level increases, and the entropy decreases. By contrast, at starting cut in the original location, decrease of entropy is not very clear, the number of mine earthquakes is small and the energy is small; this is due to slower speed and less influence at the beginning of mining, so entropy change is relatively small. It can be speculated that entropy theory can be used to predict the occurrence of mine earthquake. When entropy reduces, mine earthquake probably happens.

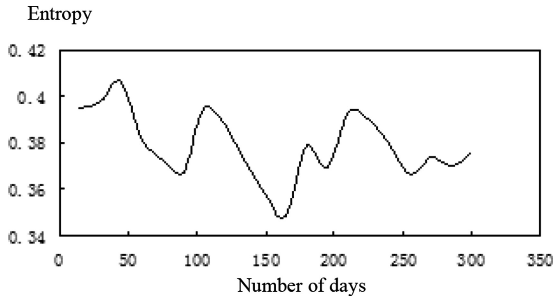


Fig. 2. The change curve of mine earthquake information entropy with time

5 Conclusion

(1) The study defines the mine earthquake information entropy, and clarifies the normal index and distribution types by using the maximum information entropy principle. The relationship among the information entropy, distribution types and standard deviation is also analyzed. The information entropy model contains G-R and fractal index model.

(2) By the actual monitoring data, the study verifies mine earthquake magnitude distribution does not accord with index distribution of G-R and fractal. It is also different from normal distribution. If these standards are applied to process data will lose useful information. That is to say, it has certain limitation to forecast mine earthquake by using the b value of G-R and fractal dimension.

(3) Through the analysis on the mine earthquake information entropy, this paper puts forward the method which can be used to predict the occurrence of mine earthquake according to relative information entropy reduction, and use mine earthquake testing result of 1305 face in Dongtan mines within nearly a year to calculate. The results show that the mine earthquake information entropy method is a feasible method, and has guidance for the practical work.

Acknowledgement. This work is supported by the National Natural Science Foundation of China, grant NO. 51774173, the Natural Science Foundation of Liaoning Province, China, grant No. 201602351 and Open Fund for the State Key Laboratory of Seismological Dynamics, grant No. LED2015B01.

References

1. Pan, Y.-S.: Study on the occurrence and damage law of mine earthquake. Post doctoral report, Institute of geology, China Earthquake Administration, Beijing (2003)
2. Pan, Y.-S., Zhao, Y,-F., Guan, F.-H., et al.: Study on rockburst monitoring and orientation system and its application. Chin. J. Rock Mechan. Eng. **26**(5), 1002–1012 (2007)
3. Lu, C.-P., Dou, L.-M., Wu, X.-R., et al.: Experimental and empirical research on frequency-spectrum evolvement rule of rockburst precursory microseismic signals of coal-rock. Chin. J. Rock Mech. Eng. **27**(3), 519–525 (2008)
4. Jiang, F.-X., Yang, S.-H., Cheng, Y,-H., et al.: A study on microseismic monitoring of rock burst in coal mine. Chin. J. Geophys. **49**(5), 1511–1516 (2006)
5. Jiang, H.-K., Diao, S.-Z.: A preliminary study on the relationship between the fractal dimension D and the b value of seismicity model with fractal structure. Acta Seismol. Sin. **17** (4), 524–527 (1995)
6. Qu, S.-X.: Relationship between fractal dimension and entropy. Chin. J. High Press. Phys. **7** (2), 127–131 (1993)
7. Singh, C., Singh, A., Chadha, R.-K.: Fractal and B mapping in eastern Himalaya and southern Tibet. World Earthq. Ser. **3**, 15–20 (2010)
8. Li, Y.-H., Liu, J.-P., Zhao, X.-D., et al.: Study on b-value and fractal dimension of acoustic emission during rock failure process. Rock Soil Mech. **30**(9), 2559–2564 (2009)
9. Xie, H.-P., Gao, F., Zhou, H.-W., et al.: Fractal study on fracture and fragmentation of rock. J. Disaster Prev. Mitig. Eng. **23**(4), 1–9 (2003)
10. Deng, G.-Z., Zhu, W.-S.: Basic characteristics of nonlinear unloading and entropy change of rock mass. J. Xi'an Min. Inst. **17**(4), 332–335 (1997)
11. Su, R.-K.: Statistical Physics. Higher Education Press, Beijing (2004)
12. Yang, Z.-R.: Quantum Statistical Physics. Higher Education Press, Beijing (2007)
13. Zhu, X.-L.: Applied Information Theory. Tsinghua University Press, Beijing (2000)
14. Fu, Z.-Y.: Information Theory Basic Theory and Application. Electronic Industry Press, Beijing (2006)
15. Miyano, H.: Identification model based on the maximum information entropy principle. J. Math. Psychol. **45**(1), 27–42 (2001)
16. Qiao, W.-P., Wang, Y.-Q.: Information entropy and error distribution fanction. Phys. Eng. **13**(1), 26–28 (2003)
17. Feng, S.-Y.: Information entropy and maximum entropy principle. Water Power Sci. Technol. **22**(3), 24–29 (1995)
18. Feng, L.-H.: The maximum entropy principle and the relation between earthquake frequency and magnitude. Seismol. Geol. **25**(2), 260–265 (2003)
19. Cheng, E.-L., Zhang, M.-F.: The application of the maximum entropy principle in the earthquake recurrence relation. J. Seismol. Res. **10**(2), 139–144 (1987)
20. Xu, C.-H., Ren, Q.-W., Li, R.: Entropy catastrophe criterion of surrounding rock stability. Chin. J. Rock Mech. Eng. **23**(12), 1992–1995 (2004)
21. Wang, H.-T., Zhu, L.-R., Yang, M.-J., et al.: Information entropy analysis of seismic precursor observation data. Earthquake **5**, 13–18 (1991)

Automated Design Thinking Oriented on Innovatively Personified Projects

Petr Sosnin(✉)

Ulyanovsk State Technical University, 432027 Ulyanovsk, Russia
sosnin@ulstu.ru

Abstract. The paper presents a version of automating the design thinking approach in its application by an inventor of the certain innovation for a transfer from an innovative intention to a tested prototype of a possible project. A similar case of starting the work with the innovation is arisen in conditions, in which the inventor wants to evaluate the conceived value without intermediaries before decision making about objectifying this value through the design process. The suggested way of automating is based on the use of question-answer interactions of the inventor with the accessible experience, registration of verbal traces of such interactions in the semantic memory and processing the traces for achieving the architectural and cause-and-effect understanding with the use of tested prototypes. Enumerated actions must be implemented in the specialized toolkit OwnWIQA that is oriented on innovatively personified projects.

Keywords: Conceptual designing · Design thinking · Question-Answering System with software

1 Introduction

In recent years, for starting the work with innovative intentions, it is popular to use the Design Thinking approach (DT-approach) and its embodiment in corresponding technologies and tools [1]. The choice of such approach is caused by organically including this kind of work at the beginning of the project lifecycle if an evaluation of outcomes of design thinking is positive and lead to the usefulness of developing the conceived object or system. In this case and this sense, implementation of design thinking corresponds to the first stage of the design process.

The inclusion of design thinking in the life cycle of the project leads to the automation of such actions of designers, and it is easy to agree that rational automation must be coordinated with the automation of other components of the design process.

At the same time, it is possible a case when innovation is invented by a human who wants to estimate conceived value without intermediaries before decision making about objectifying this value. Maybe, later, the necessary objectivation will be designed by the inventor of the innovation. Similar cases lead to Innovatively Personified Projects that require using the personified tools for such version of inventor's activity, including the implementation of the DT-approach.

In the paper, we suggest a way of automating that is based on the use of question-answer interactions of the inventor with the accessible experience, registration of verbal

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 312–323, 2018.
https://doi.org/10.1007/978-3-319-99972-2_25

traces of such interactions in the semantic memory and processing the traces for achieving the architectural and cause-and-effect understanding with the use of tested prototypes. Enumerated actions must be implemented by the designer in the personified toolkit OwnWIQA (WIQA – Working In Questions and Answers) that is oriented on conceptual designing [2], including innovatively personified designing.

2 Features of Personified Design Thinking

2.1 Applied Design Thinking

The content of this paper focusses on estimating an innovative intention in conditions when the innovator applies the DT-approach in the environment of the toolkit Own-WIQA. These conditions are schematically shown in Fig. 1.

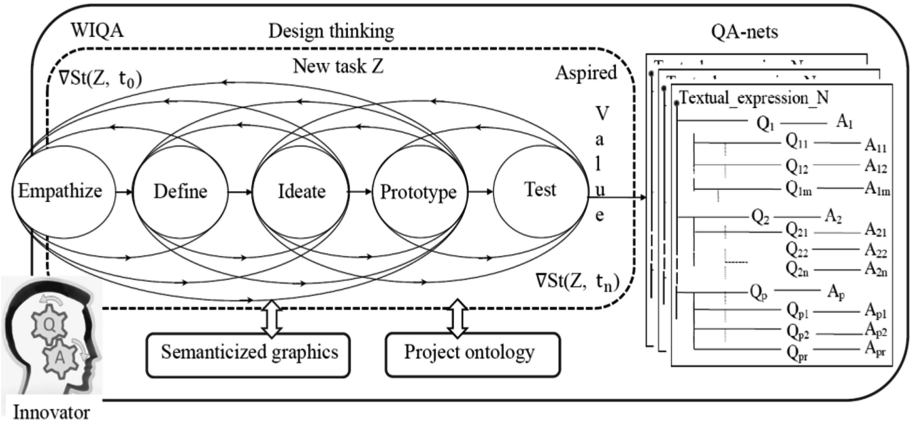


Fig. 1. The iterative process of design thinking

In our version of the DT-approach, the work with the task Z* as a root task of the conceived project begins with a list of keywords registering this task at the level of an initial uncertainty $\nabla U(Z^*, t_0)$. The initial state of the task can be presented with an implication [3]

$$??^U U(Z_i) \overset{???^W W(Z_i)}{\longrightarrow} ?^V V(Z_i),$$

where symbols $?^U$, $??^V$ and $?^{??W}$ indicates applying of question-answer approach (QA-approach) [2] to the components of a condition U, value V and potential a solution W of the task Z*.

In accordance with the QA-approach, questions are objects of interests and applications. In the implication, signs "?", "??" and "???" indicate on uncertainties of a different measure. In this place and below, these signs are used as special indicators required to extract or generate the questions during question-answer analysis of the processed text.

Thus for an innovative project, it is typical to start the life cycle of the project with uncertainty $?^{V}V(Z^{*})$, achieving of which is bound with the project goal. In the described version of work, the list of the certain keywords presents the initial point of the life cycle that corresponds to starting the work with the root project tasks Z* in the frame of design thinking. Below we present elements of such work for the definite project while here we describe the steps of design thinking only schematically.

Looking through keywords, the innovator (fulfilled the role of the designer) creates some discourses disclosing the essential expectations of the potential users (the Empathize step). These discourses contain information about important requirements, QA-analysis of which helps the designer to formulate the initial statement St(Z*, t2) of the task Z* (The Define step). Analyzing the text St(Z*, t2), the designer tries to invent the idea that can lead to the appropriate solution (the Ideate step), for estimating of which the corresponding prototype should be built (the Prototype step) and tested (the Test step).

In the general case, the described way of working has an iterative character with possible returns to previous steps. Moreover, the first solution is seldom effective, and the search for alternative solutions should be repeated for the future choice of one of them.

The scheme in Fig. 1 also includes some components that uncover conditions of realizing our version of the DT-process. It is implemented in the instrumentally modeling environment WIQA that includes a complex of means for the use of semanticized graphics and project ontology in the analysis and processing of its outcomes [4].

2.2 Achieving the Adequacy of Verbal Expressions

In any case, a lifecycle of any innovative project includes the following lines of actions:

- The choice and creatively use of an appropriate subset of the certain naturally artificial language (such actions is better to interpret as a creating the specialized language $L^{P}(t)$ of the corresponding project P(t)).
- An implicit or explicit generating the problem space that opens the possibility of building the alternative versions of objectifying the investigated intention.
- Formulating an initial statement for a root task that corresponds to the alternative version of a project.
- Substantiated prototyping of any alternative and its testing.
- The choice of a better alternative.

There is a fundamental difference between collaborative and personified implementations of these actions within the framework of design thinking for innovative intention. The main difference is related to the way of achieving the conviction that the implied and used definitions of verbal signs and descriptions of essences, objects, and situations are adequate to the conceived project.

In the collaborative achievement of conviction, each member of the group involves in the process of potential corrections of verbal expressions of definitions and descriptions. Such corrections are based on using the unique experience of each member of the group, and this form of critique and correction is additional for ways that can be used for achieving the personified conviction.

Thus, in the personified attainment of persuasion, the innovator can rely only on own experience and the use of logically instrumental means to ensure the adequacy of definitions and descriptions. These means must be compatible with mechanisms of applied design thinking, and especially those mechanisms that responsible for achieving the sufficient degree of understanding.

Among means included in the way suggested in the paper, important places occupy a question-answer analysis, conceptual space of design thinking, and substantially evolutionary theorizing, each of which contributes to the controlled detailing the innovative intention and its estimating.

2.3 The Essence of Question-Answer Approach

As told above, any application of design thinking begins with an uncertainty $\nabla U(Z^{*}, t_0)$ of a root task Z^{*} corresponding a potential project of a conceived innovation, and such uncertainty is expressed with a list of keywords. This list is an informational source for starting the generation of discourses by the designer (from this point of the paper we will use the following designation of the innovator), who registers them in textual forms. Let us assume that processing the keywords has led to the text T_j that contains the uncertainty $\nabla U(T_j, t_1) = \nabla U(Z^{*}, t_1)$.

For reducing the current uncertainty, the designer must conduct the appropriate analysis, interacting with the investigated text T_j. For such work, we have developed the QA-approach, based on the actual division of the textual expression and question-answering [4] that is schematically shown in Fig. 2.

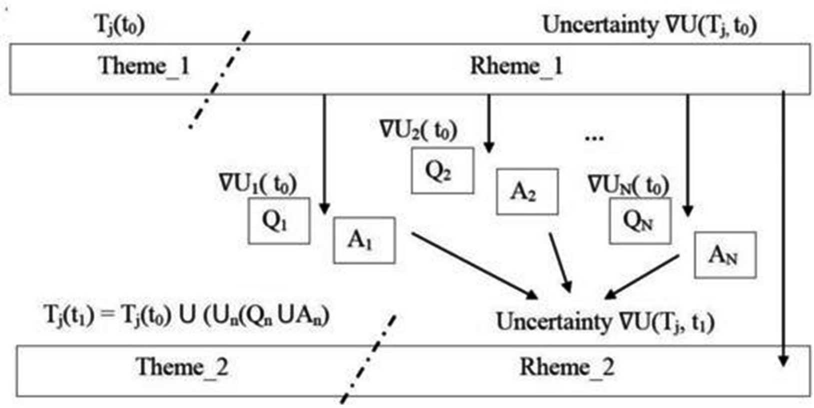


Fig. 2. Reducing the uncertainty of textual unit.

In interactions with the text, actual division as a communicative mechanism breaks up any sentence on two parts — “theme” and “rheme,” where “theme” is the predictable starting point of the communicative unit, and “rheme” indicates on new information that should receive an additional value.

In any textual unit, the designer can separate rheme and extract questions {Qn} reducing the existed uncertainty with the help of the project ontology in its current state. Descriptions of questions and corresponding answers lead to additional textual units, the content of which reduce initial uncertainty and so on.

2.4 Conceptual Space of Design Thinking

In our research and practice of conceptual designing the systems with software, we master the reflection of basic essences of the operational space onto their conceptual representations in forms of QA-nets, combining of which in the wholeness is interpreted as a conceptual space CS(t) of the corresponding project [4]. The CS(t) is an artifact that models a space for designing. Interactions of the designer with CS(t) through nodes of QA-nets are shown in Fig. 3.

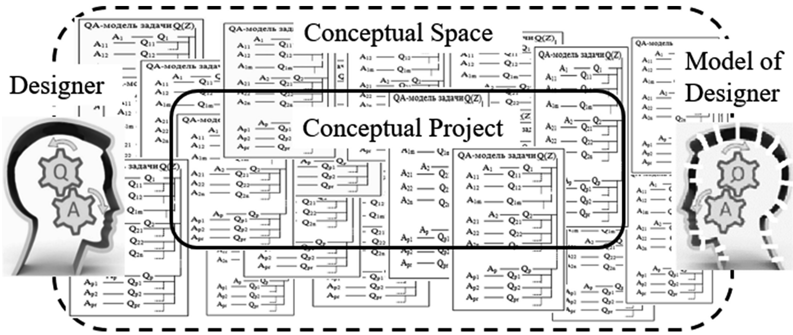


Fig. 3. Interactions of the designer with nodes of QA-nets

The scheme underlines, the conceptual project CP(t) has a representation that is located in the CS(t), and components of this representation are also objectified in forms of QA-nets. In accordance with the system approach, the CP(t) is a system located in the environment designated as the CS(t). Moreover, the CP(t) in its current state consists of components that were extracted from the CS(t) and processed by the necessary way.

In our version of the artifact CS(t), the designer begins to create this space with the “empty state,” starting with the moment of time t_0. As it was shown in Fig. 1, the designer is working in the CS-environment that is evolved on the course of design thinking. Space is opened for interacting with an embedded object, their groups, and relations. Among these relations, there are cause-and-effect regularities of the space.

Thus, the CS(t) has a potential for checking the adequacy of verbal expressions and even for correctly generating them or for extracting the expressions from the current state of the CS(t). The use of the artifact helps the designer in achieving the sufficient degree of the personified conviction in estimating the adequacy of definitions and descriptions and in their understanding.

2.5 Substantiated Theorizing

One more artifact that helps to order and check the designer' actions and their outcomes in design thinking is a substantially evolutionary theory $Th^P(t)$ of the project P(t), constructs of which are created on the base of the current state of the space CS(t). The designer creates and uses the theory $Th^P(t)$ in parallel with other actions for different purposes, among which it is accessible such as analyzing, describing, explaining, predicting and understanding.

Any applied theory $Th^P(t)$ belongs to the subclass of the Grounded theory of the constructive type, starts from the root question and evolves by extracting its constructs from facts, the role of which fulfills verbal traces of interactions of the designer with accessible experience [4]. Such traces are registered in nodes of QA-nets.

The theory $Th^P(t)$ has a phase structure that includes:

- Before-theoretical phase assembling facts about interactions with experience;
- The descriptive phase that logically combines the theory constructs formed by processing the content of the before-theoretical phase;
- Classification phase objectified in the form of the project ontology;
- Identifiedly measurable phase, components of which are outcomes of diagrammatical processing helping to understand the textual expressions.

Thus, creating and using the theory $Th^P(t)$ support achieving the sufficient degree of the personified conviction in estimating the adequacy of definitions and descriptions that, in its turn, support achieving and explicit presenting their understanding.

3 Related Works

The first group of related works concerns the subject area "Design Thinking." The informational source [5] discloses basic features of this activity in developing the systems with software. The becoming of the DT-methodology is described in the paper [1]. The interesting analysis of the nature of design thinking is conducted in the paper [3] where K. Dorst connects this nature with two types of abductive reasoning of the designer in interactions with situated problems.

The next group of related works focuses on the constructive use of conceptual spaces in the design practice. In this group, we mark the publication [6] defined the ontological viewpoint on the CS, the study [7] disclosed the CS from the viewpoint of "the construction, exploration, and expansion," and the paper [8] that suggests some templates for documenting the state of the space in the form suitable for decision-making.

One more group of publications has deals with the study of theorizing in behavior-oriented subject areas. The choice of such group was caused by behavioral nature of designing [9]. Here we mark the review [10] (published in 2013), in which the author separates the analyzed papers into three areas - flexible design, distributed development and formulating the requirements. The second review [11] (published in 2016) notes that in the vast majority of investigated publication, Grounded Theory is applied fragmentarily, either at the level of workflows or to some kinds of activity of designers.

All papers indicated in this section were used as sources of requirements in developing the set of instrumental means provided automated design thinking in the personified innovative projects.

4 Features of the Suggested Way of Design Thinking

4.1 Becoming of the Empathy Stage

Traditionally, actions of the Empathy step are started with generating some discourses that must express the important expectations of potential users of outcomes of designing.

In the described case, the designer starts to create these discourses by interacting with an open set of keywords and "translating" them in some useful memos that can be used for the following actions:

- Extracting a set of facts disclosing traces of access to the natural experience in mental space;
- Registering of these facts as potential components of the CS(t);
- Generating the useful discourses that will highly likely transform to requirements of the conceived project.

The designer will generate discourses in turn, and each generation step will be based on the following actions:

- Extracting a set of significant lexemes from the chosen discourse (from the textual unit) and preparing this set for including its elements into the project language $L^P(t)$;
- Extracting the concepts (notions) from the set of lexemes included in $L^P(t)$;
- Defining any discovered concept and including it in the project ontology $O^P(t)$;
- Discovering such relations among concepts that define normative kinds of systematization in the ontology ("is-a," "part of," "cause-and-effect" and the other relations).
- Registering the normative relations in the ontology $O^P(t)$;

In current states, outcomes of indicated actions are used for checking the controlled use of lexis in the discourse, and such checking can lead to corrections of the discourse expression. One more reason for the discourse correction is an attempt to achieve an understanding of the discourse.

All indicated actions with any formulated discourse $D_i(t)$ are implemented in the semantic area of the toolkit OwnWIQA in the frame of initial state of the task Z*, the first subtask $Z^F(Z^*, t)$ of which is normatively defined as "Formulating the statement of the task Z*(t)". In $Z^F(Z^*, t)$, each discourse $D_i(t)$ is interpreted as a source of questions that must be extracted by the designer from the text of the discourse during its question-answer analysis, the result of which has the form of the corresponding QA-net.

After fulfilling such work, the designer must generate the initial statement St(Z*, t) of the task Z*(t) as a textual wholeness, the content of which is not contradicting to any discourses. In other words, the statement St(Z*, t) is built as an abstraction of discourses, each of which is "inferred" from the statement. This part of designer's activity corresponds to the stage "Define" of the DT-approach.

The statement St(Z*, t) is a textual expression, which the designer must process as texts of discourses and also as texts of questions and answers generated during QA-analysis. The principled role in such processing fulfill the use of the ontology $O^P(t)$ and figuratively semantic support [2] of designer's actions, without which achieving the sufficient degree of understanding is problematical.

4.2 Early Steps of Design Thinking

The toolkit OwnWIQA consists of subsystems, one of which is a subsystem "Base of Precedents." So there was a time when we started the project of this subsystem. On an example of the project, we clarify some elements of our version of design thinking. Clarifying we start with the following discourse:

D1. When operating the system with the built-in complex "Base of Precedents" (?) in the event of an unpredictable situation (?) for both the designers and users, there should be an opportunity for an intelligent reaction (?) aimed at handling (?) the situation for continued actions.

The reason of formulating this discourse is caused by the real practice of developing the modern systems with software that demonstrate unpredictable influences of numerous situational factors both as on the process of designing so on the process of exploiting. In the review [12], it is indicated about 400 such situational factors are mapped into 48 of their variants distributed across 11 groups.

Some points in the text D1 include the sign "?" indicated on directions of question-answer analysis. Among units of the conducted analysis we mark the following of them:

Q1. What is meant by intellectual reactions of a designer or a user?
A1. With a new intellectual reaction, it is rational to bind the search of a new occupational action that corresponds to a way of solving a new task (?), conditions of which correspond to unpredictably arisen situation.

Answer A1 is better to interpret as the requirement that is habitual for the designer, but for the user, it is intellectual function embedded into the system to be developed.

Q2. How to specify similar reactions?
A2. Since the reaction is recommended to be built in the form of finding a solution to a new task, it is rational to represent it in the form of a precedent model (?) taking into account analogies (?) with intellectually processed (?) conditioned reflexes (with the conditioned reflexes that have the kind of human actions).

In this answer, we orient on the following definition "precedents are actions or decisions that have already happened in the past and which can be referred to and justified as an example that can be followed when the similar situation arises" [13].

The answer A2 also relates the question "What to do with the built precedent model?" For the user, there are the following versions of the reaction to the question:

- Transmitting the model of the precedent to specialists who use it for evolving the system functionality;

- Developing the precedent and embedded it into the system without intermediaries if the user has a sufficient qualification.
- Other marked signs of questions in D1 and answers A1 an A2 indicate ways of continuing the question-answer analysis of the discourse D1. Other indicated question signs in D1 and answers A1 and A2 indicate ways of continuing the question-answer analysis of the discourse D1. Among next question-answer pairs, we mark the following refinements:
- Creating the next precedent model, it needs to implement in parallel with solving the corresponding new task, beginning the processes with the use of automated design thinking;
- Applied design thinking must include the functionalities that can help the designer in conducting the thought experiments with the use of mental imagery [14].

In the project of the subsystem "Base of Precedent" was a time when we have formulated the following root task:

Z(t): It needs to develop a subsystem "Base of Precedent", operative creating (?) and using (?) of which will facilitate (?) enhancing (?) the intellectual effectiveness (?) of the designer' activity in working (?) with unpredictable project tasks in designing the systems with software that will include mechanisms (?) for similar intellectual reacting (?) for users in unpredictable situations (?) arisen in exploiting of such systems.*

Here, questions marks also indicate ways for continuing the analysis in the frame of the suggested version of automated design thinking, outcomes of which were used in the development of "Base of Precedent".

4.3 Project Ontology

In actions that are similar to the described above, the designer achieves necessary results with the use of automated interactions with the project ontology, the framework of which is presented in Fig. 4.

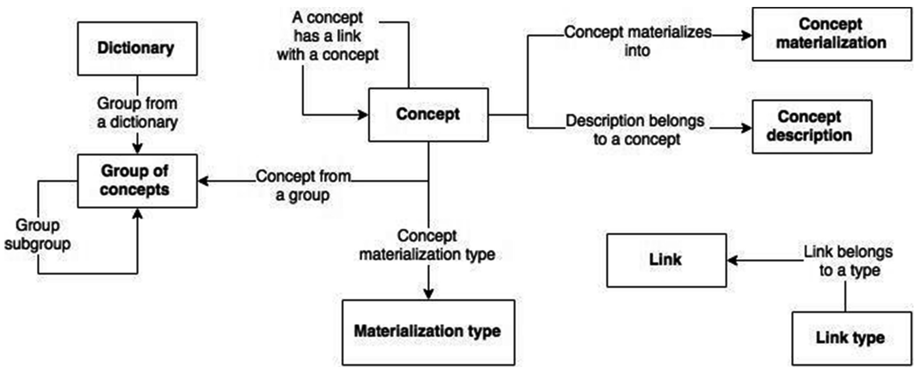


Fig. 4. The framework of the project ontology

The project ontology is created in the form of a *working dictionary* (corresponding with a certain project), which is divided into *groups* and *subgroups*. An ontology *concept* is a dictionary item that can be presented as a single word or a collocation. Concepts have their *definitions* designed to explain the meaning of a concept or to store axioms related to it. Since every concept of the project must be related to its implementation, it is necessary to classify the cases when a concept belongs to a certain part of the implementation – for that, a *materialization* component is used.

Each materialization has its type. For linking the pairs of concepts, the designer can use (choose and assign) necessary *Relations* from their normative set (*is a*; *part of*; *has a part; associative relations (similarity, consequence, contrast, time, space, synonyms) and others*). The designer can include additional relations if they will be useful in the work with the project.

Among important goals of interactions with the ontology, we mark the controlled use of lexis, the necessity of achieving the sufficient level of understanding and extracting the useful questions. Means providing the necessary ontological support are integrated into the subsystem "Project Ontology" embedded into the toolkit OwnWIQA.

4.4 Means of Systematized Graphics

For achieving and explicit expressing the sufficient degree of understanding, the designer includes the necessary concepts and graphical models to the construct St (Z*, t), and embedded components will provide reuse of understanding in the process of which these models will repeatedly activate the mental imagery. Without such activation, the needed and expected understanding is impossible. The subsystem "Figuratively Semantic Editor" of the toolkit OwnWIQA supports creating and transforming the visual models for types presented in Fig. 5.

A set of types include:

1. Pictorial type M^P, models of which help to express the architectural forms of understanding.
2. Declarative type M^D that is intended for visualizing the semantic nets of textual units, for examples, such as sentences and discourses. The main reason for using this type of visualizing is the check of $St(Z_i, t)$ and its verbal components via mechanisms of declarative programming.

Conceptually algorithmic type MA provides presenting cause-and-effects forms of understanding. For the creation of such form, the designer can use a pseudocode language L^{WIQA} that is adjusted on the semantic memory of the toolkit OwnWIQA. In the current state of WIQA, the type M^A supports the work with Use-Case diagrams, Activity diagrams, and diagrams of Classes. For the transition to the version P^A, it is applied the automatic mode based on the model-driven approach [14].

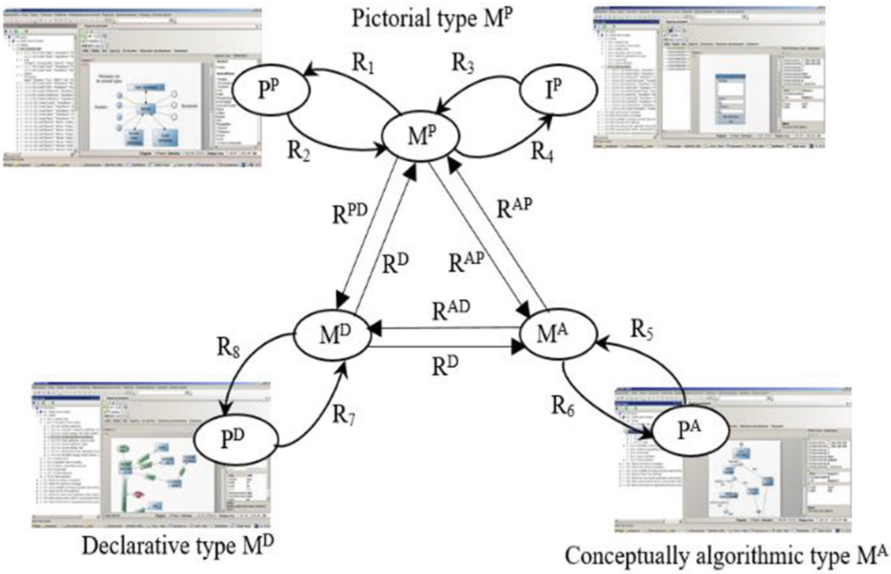


Fig. 5. Normative types of visual models

5 Conclusion

The paper has deals with the application of the DT-approach in conditions, in which the author of the certain innovation tries to estimate the conceived value without intermediaries before decision making about objectifying this value with intermediaries, for example, with a team of specialists. In other words, the innovator tries to build the very simplified version of the innovatively personified project.

For these cases, we developed the automated version of the DT-approach, in which the quality of innovator's actions are defined by using such artifacts as the conceptual space CS(t), project language $L^P(t)$ and project theory $Th^P(t)$ that are created in coordination and in parallel with other actions of design thinking, the kernel of which is the use of question-answer analysis applied to the generated in registered textual units.

Such personified activity must be realized in the toolkit OwnWIQA that is intended for conceptual designing accompanied by creating and using of indicated artifacts with intensive use subsystems "Project Ontology" and "Figuratively Semantic Editor".

Acknowledgement. This work was supported by the Russian Fund for Basic Research (RFBR), Grant #18- 07-00989a, Grant # 18-47-73001r-a, and the State Contract №. 2.1534.2017/4.6.

References

1. Leifer, L., Meinel, C.: Manifesto: design thinking becomes foundational. In: Plattner, H., Meinel, C., Leifer, L. (eds.) Design Thinking Research. UI, pp. 1–4. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-19641-1_1
2. Sosnin, P.: Experience-Based Human-Computer Interactions: Emerging Research and Opportunities. IGI-Global, Hershey (2018)
3. Dorst, K.: The nature of design thinking, in DTRS8 interpreting design thinking. In: Proceeding of Design Thinking Research Symposium, pp. 131–139 (2010, 2013)
4. Sosnin, P.: Substantially evolutionary theorizing in designing software-intensive systems. Information **9**(4), 1–29 (2018)
5. Häger, F., Kowark T., Krüger, J., Vetterli Ch., Übernickel, F., Uflacker, M.: DT@Scrum: integrating design thinking with software development processes, design thinking. In: Understanding Innovation, pp. 263–289 (2014)
6. Gärdenfors, P.: Semantics based on conceptual spaces. In: Banerjee, M., Seth, A. (eds.) ICLA 2011. LNCS (LNAI), vol. 6521, pp. 1–11. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-18026-2_1
7. Heape, C.: The design space: the design process as the construction, exploration, and expansion of a conceptual space. Ph.D. dissertion. The University of Southern Denmark, Sønderborg, Denmark (2007)
8. Dove, G., Hansen, N.B., Halskov, K.: An argument for design space reflection. In: Proceedings of the 9th Nordic Conference on Human-Computer Interaction (2016)
9. Charmaz, K.: Constructing Grounded Theory, 2nd edn. Sage, London (2014)
10. Badreddin, O.: Thematic review and analysis of grounded theory application in software engineering. In: Advanced Software Engineering (2013)
11. Stol, K., Ralph, P., Fitzgerald, B.: Grounded theory research in software engineering: a critical review and guidelines. In: Proceedings of the 2016 International Conference on Software Engineering, Austin, TX, USA, 14–22 May 2016, pp. 120–131 (2016)
12. Clarke, P., O'Connor, R.V.: The situational factors that affect the software development process: towards a comprehensive reference framework. J. Inf. Softw. Technol. **54**(5), 433–447 (2012)
13. Precedent: http://dictionary.reference.com/browse/precedent (2018)
14. Sosnin, P., Galochkin, M.: Way of coordination of visual modeling and mental imagery in conceptual solution of project task. In: Benferhat, S., Tabia, K., Ali, M. (eds.) IEA/AIE 2017. LNCS (LNAI), vol. 10350, pp. 635–638. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-60042-0_70

A Comparison of Concept and Global Probabilistic Approximations Based on Mining Incomplete Data

Patrick G. Clark[1], Cheng Gao[1], Jerzy W. Grzymala-Busse[1,2(✉)], Teresa Mroczek[2], and Rafal Niemiec[2]

[1] Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, KS 66045, USA
patrick.g.clark@gmail.com, {cheng.gao,jerzy}@ku.edu
[2] Department of Expert Systems and Artificial Intelligence, University of Information Technology and Management, 35-225 Rzeszow, Poland
{tmroczek,rniemiec}@wsiz.rzeszow.pl

Abstract. We discuss incomplete data sets with two interpretations of missing attribute values, lost values and "do not care" conditions. For data mining we use two probabilistic approximations, concept and global. Concept probabilistic approximations are well known while global probabilistic approximations are introduced in this paper. The rationale for introducing global probabilistic approximations is a common opinion of the rough set community that global probabilistic approximations, as closer to the approximated concept, should be more successful. Surprisingly, results of our experiments show that the error rate evaluated by ten-fold cross validation is smaller for concept probabilistic approximations than for global probabilistic approximations.

Keywords: Incomplete data mining · Characteristic sets
Rough set theory · Probabilistic approximations.

1 Introduction

We discuss incomplete data sets with two interpretations of missing attribute values, lost values and "do not care" conditions. Both interpretations were discussed in [6]. Lost values were originally known but actually are erased or forgotten. In mining data sets with lost values we try to use only existing, specified attribute values. "Do not care" conditions may indicate a refusal to answer a question. Mining incomplete data with "do not care" conditions is based on assumption that a missing attribute value may be any value from the attribute domain.

In this paper, for data mining we use probabilistic approximations, an extension of lower and upper approximations known from rough set theory. A probabilistic approximation is defined by using an additional parameter, denoted by α and interpreted as a probability. Lower approximations are probabilistic

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 324–335, 2018.
https://doi.org/10.1007/978-3-319-99972-2_26

approximations with $\alpha = 1$, upper approximations are probabilistic approximations with α only slightly greater than zero. Probabilistic approximations, restricted to complete data sets, i.e., data sets without missing attribute values, were extensively studied in [13–21]. Probabilistic approximations were generalized to incomplete data sets in [8].

For complete data sets, a probabilistic approximation is unique for the fixed value of the parameter α. For incomplete data sets, i.e., data sets in which some attribute values are missing, there exist many possible definitions of probabilistic approximations [4,9]. In this paper we study two such approximations. The first, called a concept probabilistic approximation, is well known [2,3]. The second, introduced in this paper, is called a global probabilistic approximation. Special cases of the global probabilistic approximation, called lower and upper, were discussed in [11]. A rationale for introducing global probabilistic approximations is a common opinion of the rough set community that lower approximation should be as large as possible while the upper approximation should be as small as possible. In general, global probabilistic approximations are close to the concept to be approximated.

The main objective of this paper is to compare concept and global probabilistic approximations in terms of an error rate evaluated as a result of stratified ten-fold cross validation on some benchmark data sets. Our secondary objective is to compare both interpretations of missing attribute values, lost values and "do not care" conditions, using the same criterion. Surprisingly, using Friedman test (5% significance level), we conclude that concept probabilistic approximations are better than global probabilistic approximations (an error rate for the same data set and the same type of interpretation of missing attribute values is smaller for concept probabilistic approximations than for global probabilistic approximations). Data mining is an experimental area, results of experiments are decisive. On the other hand, an error rate for data sets with lost values and "do not care" conditions do not differ significantly.

2 Incomplete Data Sets

Table 1 presents an example of the incomplete data set. Lost values and "do not care" conditions are denoted by symbols of "?" and "*", respectively. The set of all cases will be denoted by U. In our example, $U = \{1, 2, 3, 4, 5, 6, 7, 8\}$. The set of all cases with the same decision value is called a *concept*. For example, the set $\{1, 2\}$ is a concept. We say that $a(x) = v$ if an attribute a has value v for a case x. For example, $Temperature(1) = normal$.

For completely specified data sets, for an attribute-value pair $(a,\ v)$, a *block* of $(a,\ v)$, denoted by $[(a,\ v)]$, is defined as follows

$$[(a,\ v)] = \{x | x \in U, a(x) = v\}.$$

Table 1. An incomplete data set

Case	Attributes			Decision
	Temperature	Wind	Humidity	Trip
1	normal	low	high	yes
2	*	high	*	yes
3	high	?	low	no
4	very-high	high	high	no
5	?	low	low	no
6	high	high	high	no
7	high	?	high	no
8	normal	high	low	no

For incomplete decision tables the definition of a block of an attribute-value pair was modified in [6,7] in the following way

– If for an attribute a and a case x we have $a(x) = ?$, the case x should not be included in any blocks $[(a, v)]$ for all values v of attribute a,
– If for an attribute a and a case x we have $a(x) = *$, the case x should be included in blocks $[(a, v)]$ for all specified values v of attribute a.

For our example from Table 1, all blocks of attribute-value pairs are
$[(Temperature, normal)] = \{1, 2, 8\}$,
$[(Temperature, high)] = \{2, 3, 6, 7\}$,
$[(Temperature, very - high)] = \{2, 4\}$,
$[(Wind, low)] = \{1, 5\}$,
$[(Wind, high)] = \{2, 4, 6, 8\}$,
$[(Humidity, low)] = \{2, 3, 5, 8\}$,
$[(Humidity, high)] = \{1, 2, 4, 6, 7\}$.

The *characteristic set* $K_B(x)$ is defined as the intersection of the sets $K(x, a)$, for all $a \in B$, where $x \in U$, B is a subset of the set A of all attributes and the set $K(x, a)$ is defined as follows

– If $a(x)$ is specified, then $K(x, a)$ is the block $[(a, a(x))]$ of attribute a and its value $a(x)$,
– If $a(x) = ?$ or $a(x) = *$, then $K(x, a) = U$.

For the data set from Table 1 and $B = A$, the characteristic sets are
$K_A(1) = \{1\}$,
$K_A(2) = \{2, 4, 6, 8\}$,
$K_A(3) = \{2, 3\}$,
$K_A(4) = \{2, 4\}$,
$K_A(5) = \{5\}$,
$K_A(6) = \{2, 6\}$,

$K_A(7) = \{2, 6, 7\}$,
$K_A(8) = \{2, 8\}$.

3 Probabilistic Approximations

Three kinds of probabilistic approximations, called singleton, subset and concept, were introduced in [1].

3.1 Singleton, Subset and Concept Approximations

A B-singleton probabilistic approximation of X with the threshold α, $0 < \alpha \leq 1$, denoted by $appr_{\alpha,B}^{singleton}(X)$, is defined by

$$\{x \mid x \in U,\ Pr(X \mid K_B(x)) \geq \alpha\},$$

where $Pr(X \mid K_B(x)) = \frac{|X \cap K_B(x)|}{|K_B(x)|}$ is the conditional probability of X given $K_B(x)$ and $|Y|$ denotes the cardinality of set Y. A B-subset probabilistic approximation of the set X with the threshold α, $0 < \alpha \leq 1$, denoted by $appr_{\alpha,B}^{subset}(X)$, is defined by

$$\cup\{K_B(x) \mid x \in U,\ Pr(X \mid K_B(x)) \geq \alpha\}.$$

A B-concept probabilistic approximation of the set X with the threshold α, $0 < \alpha \leq 1$, denoted by $appr_{\alpha,B}^{concept}(X)$, is defined by

$$\cup\{K_B(x) \mid x \in X,\ Pr(X \mid K_B(x)) \geq \alpha\}.$$

For Table 1 and the concept $\{1, 2\}$, all conditional probabilities $Pr(X|K_A(x)$, where X is a concept, are presented in Tables 2 and 3.

Table 2. Conditional probabilities $Pr([(Trip, yes)]|K_A(x))$, I

x	1	2	3	4
$K_A(x)$	{1}	{2, 4, 6, 8}	{2, 3}	{2, 4}
$Pr(\{1,2\} \mid K_A(x))$	1	0.25	0.5	0.5

Table 3. Conditional probabilities $Pr([(Trip, yes)]|K_A(x))$, II

x	5	6	7	8
$K_A(x)$	{5}	{2, 6}	{2, 6, 7}	{2, 8}
$Pr(\{1,2\} \mid K_A(x))$	0	0.5	0.333	0.5

All distinct probabilistic approximations are

$appr_{0.25}^{singleton}(\{1, 2\}) = \{1, 2, 3, 4, 6, 7, 8\}$,

$appr_{0.333}^{singleton}(\{1, 2\}) = \{1, 3, 4, 6, 7, 8\}$,

$appr_{0.5}^{singleton}(\{1, 2\}) = \{1, 3, 4, 6, 8\}$,

$appr_{1}^{singleton}(\{1, 2\}) = \{1\}$,

$appr_{0.333}^{subset}(\{1, 2\}) = \{1, 2, 3, 4, 6, 7, 8\}$,

$appr_{0.5}^{subset}(\{1, 2\}) = \{1, 2, 3, 4, 6, 8\}$,

$appr_{1}^{subset}(\{1, 2\}) = \{1\}$,

$appr_{0.25}^{concept}(\{1, 2\}) = \{1, 2, 4, 6, 8\}$,

$appr_{1}^{concept}(\{1, 2\}) = \{1\}$,

$appr_{0.5}^{singleton}(\{3, 4, 5, 6, 7, 8\}) = \{2, 3, 4, 5, 6, 7, 8\}$,

$appr_{0.667}^{singleton}(\{3, 4, 5, 6, 7, 8\}) = \{2, 5, 7\}$,

$appr_{0.75}^{singleton}(\{3, 4, 5, 6, 7, 8\}) = \{2, 5\}$,

$appr_{1}^{singleton}(\{3, 4, 5, 6, 7, 8\}) = \{5\}$,

$appr_{0.5}^{subset}(\{3, 4, 5, 6, 7, 8\}) = \{2, 3, 4, 5, 6, 7, 8\}$,

$appr_{0.667}^{subset}(\{3, 4, 5, 6, 7, 8\}) = \{2, 4, 5, 6, 7, 8\}$,

$appr_{0.75}^{subset}(\{3, 4, 5, 6, 7, 8\}) = \{2, 4, 5, 6, 8\}$,

$appr_{1}^{subset}(\{3, 4, 5, 6, 7, 8\}) = \{5\}$,

$appr_{0.5}^{concept}(\{3, 4, 5, 6, 7, 8\}) = \{2, 3, 4, 5, 6, 7, 8\}$,

$appr_{0.667}^{concept}(\{3, 4, 5, 6, 7, 8\}) = \{2, 5, 6, 7\}$, and

$appr_{1}^{concept}(\{3, 4, 5, 6, 7, 8\}) = \{5\}$.

3.2 Global Probabilistic Approximations

An idea of the global approximation, restricted to only lower and upper approximations, was introduced in [11,12]. In this paper we introduce a global probabilistic approximation, associated with a parameter α, $0 < \alpha \leq 1$. Let $B \subseteq A$ and $X \subseteq U$. A *B-global probabilistic approximation* of the concept X, based on characteristic sets, with the parameter α and denoted by $appr^{global}_{\alpha,B}$ is defined as the following set

$$\bigcup\{K_B(x) \mid \exists\, Y \subseteq U\ \forall\, x \in Y,\ Pr(X|K_B(x))\ \geq \alpha\}.$$

In general, for given sets B and X and the parameter α, there exists many B-global probabilistic approximations of X. Additionally, computing all such approximations is of exponential computational complexity. Therefore, we decided to use a heuristic version of the definition of B-global probabilistic approximation, called a MLEM2 (Modified Learning from Examples Module, version 2) B-global probabilistic approximation of the concept X, associated with a parameter α and denoted by $appr^{mlem2}_{\alpha,B}$. This definition is based on the rule induction algorithm MLEM2 [5]. The approximation $appr^{mlem2}_{\alpha,B}$ is constructed from characteristic sets $K_B(y)$, the most relevant to the concept X, i.e., with $|X \cap K_B(y)|$ as large as possible and $Pr(X|K_B(y)) \geq \alpha$, where $y \in U$. If more than one characteristic set $K_B(y)$ satisfies both conditions, we pick the characteristic set $K_B(y)$ with the largest $Pr(X|K_B(y))$. If this criterion ends up with a tie, a characteristic set is picked up heuristically, as the first on the list.

In this paper we will study only MLEM2 B-global probabilistic approximations based on characteristic sets, with $B = A$, and calling them, for simplicity, *global probabilistic approximations* associated with the parameter α. Once the global probabilistic approximations associated with α are constructed, rule sets are induced using the rule induction algorithm based on another parameter, also interpreted as a probability, and denoted by β. This algorithm also uses MLEM2 principles [9].

MLEM2 rule induction algorithm
input: a set X (a subset of U) and a parameter β,
output: a set $\mathcal{T}$ of the set X,
begin
    $G := X$;
    $D := X$;
    $\mathcal{T} := \emptyset$;
    $\mathcal{J} := \emptyset$;
    while $G \neq \emptyset$
        begin
            $T := \emptyset$;
            $T_s := \emptyset$;
            $T_n := \emptyset$;
            $T(G) := \{t \mid [t] \cap G \neq \emptyset\}$;
            while ($T = \emptyset$ **or** $[T] \not\subseteq D$) **and** $T(G) \neq \emptyset$

begin
select a pair $t = (a_t, v_t) \in T(G)$ such that
$|[t] \cap G|$ is maximum; if a tie occurs, select
a pair $t \in T(G)$ with the smallest cardinality
of $[t]$; if another tie occurs, select first pair;
$T := T \cup \{t\}$;
$G := [t] \cap G$;
$T(G) := \{t \mid [t] \cap G \neq \emptyset\}$;
if a_t is symbolic {let V_{a_t} be the domain of a_t}
then
$T_s := T_s \cup \{(a_t, v) \mid v \in V_{a_t}\}$
else {a_t is numerical, let $t = (a_t, u..v)$}
$T_n := T_n \cup \{(a_t, x..y) \mid$ disjoint $x..y$
and $u..v\} \cup \{(a_t, x..y) \mid x..y \supseteq u..v\}$;
$T(G) := T(G) - (T_s \cup T_n)$;
end {while};
if $Pr(X \mid [T]) \geq \beta$
then
begin
$D := D \cup [T]$;
$\mathcal{T} := \mathcal{T} \cup \{T\}$;
end {then}
else $\mathcal{J} := \mathcal{J} \cup \{T\}$;
$G := D - \cup_{S \in \mathcal{T} \cup \mathcal{J}} [S]$;
end {while};
for each $T \in \mathcal{T}$ **do**
for each numerical attribute a_t with $(a_t, u..v) \in T$ **do**
while (T contains at least two different
pairs $(a_t, u..v)$ and $(a_t, x..y)$ with
the same numerical attribute a_t)
replace these two pairs with a new pair
(a_t, common part of $(u..v)$ and $(x..y)$);
for each $t \in T$ **do**
if $[T - \{t\}] \subseteq D$ **then** $T := T - \{t\}$;
for each $T \in \mathcal{T}$ **do**
if $\cup_{S \in (\mathcal{T} - \{T\})} [S] = \cup_{S \in \mathcal{T}} [S]$ **then** $\mathcal{T} := \mathcal{T} - \{T\}$;
end {procedure}.

All distinct global probabilistic approximations are

$appr_{0.5}^{mlem2}(\{1, 2\}) = \{1, 2, 3\}$,

$appr_{1}^{mlem2}(\{1, 2\}) = \{1\}$,

$appr_{0.5}^{mlem2}(\{3, 4, 5, 6, 7, 8\}) = \{2, 3, 4, 5, 6, 7, 8\}$,

$appr_{0.667}^{mlem2}(\{3,4,5,6,7,8\}) = \{2,4,5,6,7,8\}$,

$appr_{0.75}^{mlem2}(\{3,4,5,6,7,8\}) = \{2,4,5,6,8\}$ and

$appr_{1}^{mlem2}(\{3,4,5,6,7,8\}) = \{5\}$.

It is obvious that $appr_{0.5}^{mlem2}(\{1,2\})$ better approximates the concept {1, 2} than $appr_{0.25}^{concept}(\{1,2\})$.

3.3 Definability

Any union of characteristic sets $K_B(x)$ is called *B-globally definable* [11]. An A-globally definable set is called *globally definable*. Let T be a set of attribute-value pairs, where all involved attributes are distinct and are members of a set B. Such set T is called *B-complex*. A block of a B-complex T, denoted by $[T]$, is the set $\cap\{[t]|t \in T\}$. Any union of blocks of B-complexes is called B-locally definable [11]. A-locally definable set is called *locally definable*.

Rules are expressed by attribute-value pairs, a rule is the following expression

$$(attribute-1, values-1)\ \&\ (attribute-2, value-2)$$
$$\&\ \ldots\ \&\ (attribute-n, value-n) \rightarrow (decision, value).$$

Any set X may be described by rules if it is locally definable, as was explained in [10]. Concept, subset and global probabilistic approximations are globally definable. However, singleton probabilistic approximations, in general, are not even locally definable. For example, for the data set from Table 1, $appr_{0.333}^{singleton}(\{1,2\})$ and $appr_{0.5}^{singleton}(\{1,2\})$ are not locally definable, since the case 8 occurs in three possible blocks of attribute-value pairs: $[(Temperature, normal)]$, $[(Wind, high)]$ and $[(Humidity, low)]$. All three blocks contain also case 2, so any intersection of these blocks contains both 2 and 8, however, 2 is not a member of both above singleton probabilistic approximations. Thus, singleton probabilistic approximations should not be used for data mining.

4 Experiments

We conducted our experiments on eight completely specified data sets obtained from the *Machine Learning Repository*, University of California at Irvine. We randomly replaced 35% of the existing, specified attribute values by question marks, indicating lost values. Then we created new data sets with "do not care" conditions by replacing question marks with asterisks.

Our data mining was based on rule induction technique. For this purpose we used the MLEM2 algorithm [5], a component of the system LERS (Learning from Examples based on Rough Sets). For evaluation of the error rate we used a single run of stratified ten-fold cross validation.

In our experiments four strategies for mining incomplete data, with lost values and "do not care" conditions, were used, since we combined these two interpretations of missing attribute values with two kinds of probabilistic approximations, concept and global. Parameter β, used for rule induction, was fixed and equal to one. Results of our experiments are presented in Figs. 1, 2, 3, 4, 5, 6, 7 and 8. In these figures, "Concept" means a concept probabilistic approximation, "Global" means a global probabilistic approximation, "?" means a lost value and "*" means a "do not care" condition.

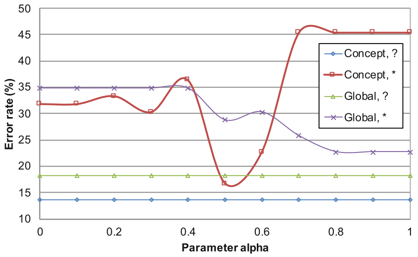


Fig. 1. Error rate for the *bankruptcy* data set

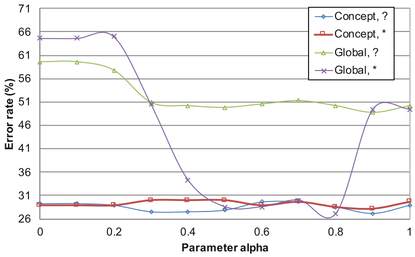


Fig. 2. Error rate for the *breast cancer* data set

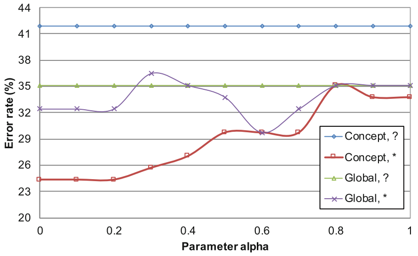


Fig. 3. Error rate for the *echocardiography* data set

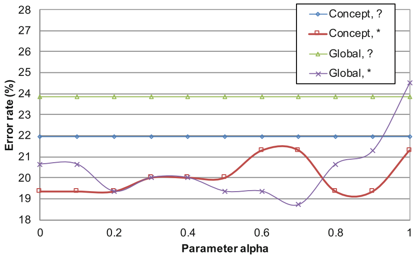


Fig. 4. Error rate for the *hepatitis* data set

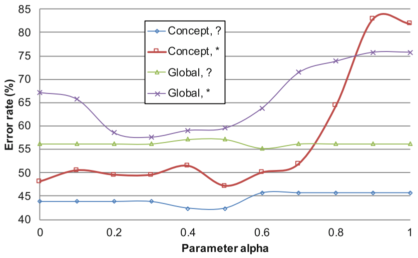


Fig. 5. Error rate for the *image segmentation* data set

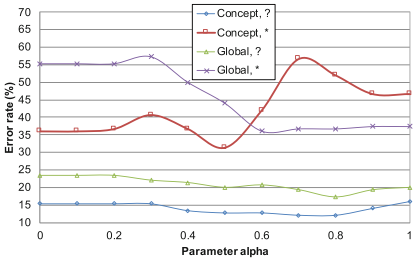


Fig. 6. Error rate for the *iris* data set

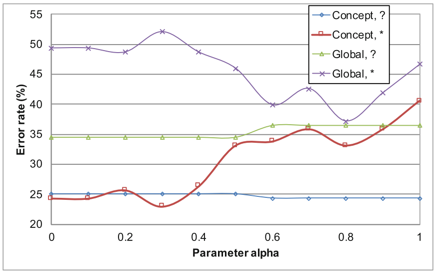


Fig. 7. Error rate for the *lymphography* data set

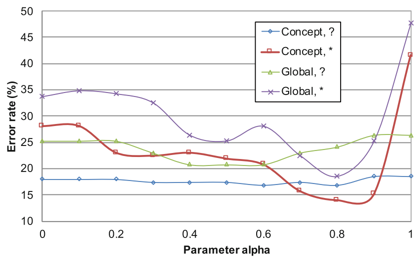


Fig. 8. Error rate for the *wine recognition* data set

To compare these four strategies we applied the Friedman rank sum test combined with multiple comparisons, with a 5% level of significance. In general, for all eight data sets the hypothesis H_0 that all four strategies are equivalent was rejected. Then we used the post-hoc test (distribution-free multiple comparisons based on Friedman rank sums) to show the differences between all four strategies.

First we are going to discuss results of the Friedman test on data sets with the same interpretation of missing attribute values. The strategy (Concept, ?) is associated with significantly smaller error rate than the strategy (Global, ?) for three data sets (*breast cancer*, *image segmentation* and *lymphography*), i.e., the strategy (Concept ?) is better than the strategy (Global, ?) for these three data sets. The strategy (Concept, *) is better than the strategy (Global, *) for two data sets (*lymhography* and *wine recognition*). There is no data set with the strategy (Global, ?) better than the strategy (Concept, ?) or with the strategy (Global, *) better than the strategy (Concept, *).

In the next analysis we compare the same type of approximation with different interpretations of missing attribute values. The strategy (Global, ?) is better than the strategy (Global, *) for two data sets (*bankruptcy* and *iris*) while the strategy (Concept, ?) is better than the strategy (Concept, *) also for two data sets (again, *bankruptcy* and *iris*). On the other hand, the strategy (Concept, *) is better than the strategy (Concept, ?) for two data sets (*breast cancer* and *hepatitis*) and the strategy (Global, *) is better than the strategy (Global, ?) on one data set (*hepatitis*). So it seems that the choice of the missing attribute value interpretation does not matter.

Finally, we compared the remaining ten strategies showing a significant difference in quality, again, in terms of an error rate. Our results show that the strategy (Concept, *) is better than the strategy (Global, ?) for four data sets (*breast cancer*, *echocardiogram*, *hepatitis* and *wine recognition*), the strategy (Concept, ?) is better than the strategy (Global, *) for five data sets (*bankruptcy*, *image recognition*, *iris*, *lymphography* and *wine recognition*). There is only one data set (*echocardiography*) for which the strategy (Global, *) is better than the strategy (Concept, ?). Taking into account the difference in interpretation of missing attribute values, it is not quite clear if the difference is due to interpretation of missing attribute values or to difference in kind of approximations.

5 Conclusions

In our experiments we investigated four strategies to mining incomplete data sets, combining two interpretations of missing attribute values, lost values and "do not care" conditions, with two types of probabilistic approximations, concept and global. As follows from our results, our final conclusion is that the concept approximation is more successful than global approximation in mining incomplete data sets, where the quality is based on the error rate. Additionally, a choice between the two interpretations of missing attribute values, lost values and "do not care" conditions is not that important. Obviously, further experimental research, with more data sets, is required.

References

1. Clark, P.G., Grzymala-Busse, J.W.: Experiments on probabilistic approximations. In: Proceedings of the 2011 IEEE International Conference on Granular Computing, pp. 144–149 (2011)
2. Clark, P.G., Grzymala-Busse, J.W.: Experiments on rule induction from incomplete data using three probabilistic approximations. In: Proceedings of the 2012 IEEE International Conference on Granular Computing, pp. 90–95 (2012)
3. Clark, P.G., Grzymala-Busse, J.W.: An experimental comparison of three interpretations of missing attribute values using probabilistic approximations. In: Proceedings of the 14-th International Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, pp. 77–86 (2013)
4. Clark, P.G., Grzymala-Busse, J.W., Rzasa, W.: Mining incomplete data with singleton, subset and concept approximations. Inf. Sci. **280**, 368–384 (2014)
5. Grzymala-Busse, J.W.: MLEM2: a new algorithm for rule induction from imperfect data. In: Proceedings of the 9th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, pp. 243–250 (2002)
6. Grzymala-Busse, J.W.: Rough set strategies to data with missing attribute values. In: Notes of the Workshop on Foundations and New Directions of Data Mining, in Conjunction with the Third International Conference on Data Mining, pp. 56–63 (2003)
7. Grzymala-Busse, J.W.: Three approaches to missing attribute values–a rough set perspective. In: Proceedings of the Workshop on Foundation of Data Mining, in Conjunction with the Fourth IEEE International Conference on Data Mining, pp. 55–62 (2004)
8. Grzymala-Busse, J.W.: Generalized parameterized approximations. In: Proceedings of the 6-th International Conference on Rough Sets and Knowledge Technology, pp. 136–145 (2011)
9. Grzymala-Busse, J.W., Clark, P.G., Kuehnhausen, M.: Generalized probabilistic approximations of incomplete data. Int. J. Approximate Reasoning **132**, 180–196 (2014)
10. Grzymala-Busse, J.W., Mroczek, T.: Definability in mining incomplete data. In: Proceedings of the 20-th International Conference on Knowledge Based and Intelligent Information and Engineering Systems, pp. 179–186 (2016)

11. Grzymala-Busse, J.W., Rzasa, W.: Local and global approximations for incomplete data. In: Proceedings of the Fifth International Conference on Rough Sets and Current Trends in Computing, pp. 244–253 (2006)
12. Grzymala-Busse, J.W., Rzasa, W.: Local and global approximations for incomplete data. Trans. Rough Sets **8**, 21–34 (2008)
13. Grzymala-Busse, J.W., Ziarko, W.: Data mining based on rough sets. In: Wang, J. (ed.) Data Mining: Opportunities and Challenges, pp. 142–173. Idea Group Publ., Hershey (2003)
14. Pawlak, Z., Skowron, A.: Rough sets: some extensions. Inf. Sci. **177**, 28–40 (2007)
15. Pawlak, Z., Wong, S.K.M., Ziarko, W.: Rough sets: probabilistic versus deterministic approach. Int. J. Man Mach. Stud. **29**, 81–95 (1988)
16. Ślęzak, D., Ziarko, W.: The investigation of the Bayesian rough set model. Int. J. Approximate Reasoning **40**, 81–91 (2005)
17. Wong, S.K.M., Ziarko, W.: INFER–an adaptive decision support system based on the probabilistic approximate classification. In: Proceedings of the 6-th International Workshop on Expert Systems and their Applications, pp. 713–726 (1986)
18. Yao, Y.Y.: Probabilistic rough set approximations. Int. J. Approximate Reasoning **49**, 255–271 (2008)
19. Yao, Y.Y., Wong, S.K.M.: A decision theoretic framework for approximate concepts. Int. J. Man Mach. Stud. **37**, 793–809 (1992)
20. Ziarko, W.: Variable precision rough set model. J. Comput. Syst. Sci. **46**(1), 39–59 (1993)
21. Ziarko, W.: Probabilistic approach to rough sets. Int. J. Approximate Reasoning **49**, 272–284 (2008)

A Study in Granular Computing: Homogenous Granulation

Krzysztof Ropiak(✉) and Piotr Artiemjew

Faculty of Mathematics and Computer Science,
University of Warmia and Mazury in Olsztyn, Olsztyn, Poland
{kropiak,artem}@matman.uwm.edu.pl

Abstract. This paper is presenting a new method of decision systems granulation in the family of methods inspired by Polkowski standard granulation algorithm. The new method is called homogenous granulation. The idea is to create the granules around each training object separately by selecting smallest r-indiscernibility ratio, based on which granule consists of group of objects with the same class. This is natural idea, where the indiscernibility level is extended until indiscernibility class contains uniform group of objects. After granulation process we have used random choice for covering of universe of objects and majority voting to create granular reflections of selected granules. The main advantage of this method is lack of necessity to estimate optimal granulation radius. We have performed experiments on data from UCI repository using 5 times cross validation 5 model. First results of homogenous granulation, in the terms of classification accuracy, are comparable with the ones of already presented algorithms with significant reduction of training data size after granulation.

Keywords: Homogenous granulation · Rough sets
Decision systems · Classification

1 Introduction

Granular rough computing is the recent paradigm in which one deals with granules, the group of objects put together by some form of similarity. There are many techniques developed in this area in recent years, among other techniques for approximation of decision systems. The main idea is lowering the size of data with maintenance of the internal knowledge for classification or regression. In the era of processing large data sets these techniques can play a significant role. One of the family of granulation methods was proposed by Polkowski in [6]. In the works, Artiemjew [1,2], Polkowski [5–9], and, Polkowski and Artiemjew [15–17] we have the presentation of standard granulation, concept dependent and layered granulation in the context of data reduction, missing values absorbtion and usage in the classification process.

The motivation to conduct this research was to take into account an idea of extension of r-insiscernibility ratio around selected object to surround it with a

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 336–346, 2018.
https://doi.org/10.1007/978-3-319-99972-2_27

set of homogenous objects in the sense of belongingness to decision class. Our new results obtained in the paper show a significant difference of this method in comparison with existing methods. First of all the indiscernibility ratio for particular central object is chosen in automatic way, by extension of ratio until the set of objects is homogenous. This particular feature is solving the problem of optimal radii search and the granulation process is performed only once. The approximation level for checked data sets is reduced by up to 50 percent in training data size with satisfactory maintenance of internal knowledge - expressed by efficiency in the classification process.

In this work we have performed preliminary research and for the sake of simplicity the results are prepared for nominal data - numeric values are treated as symbols.

The rest of the paper has the following content. In Sect. 1 we have introduction. In Sect. 2 the theoretical background. In Sect. 3 we show the detailed description of our new granulation method. In Sect. 4 we present description of the classifier used in experimental part. In Sect. 5 there are the results of the experiments, and we conclude the paper in Sect. 6.

The are three basic steps of the granulation process, the granules are computed for each training object, training data set is covered using selected strategy, and in the last step we use majority voting to get granular reflection of training system.

In the next Section, we describe the first step of mentioned procedure.

2 Granular Rough Inclusions in the Nutshell

Theoretical background of rough inclusions can be found in Polkowski [3–5], [8,9], a detailed discussion may be found in Polkowski [10].

The standard rough inclusion μ is defined as

$$\mu(v,u,r) \Leftrightarrow \frac{|IND(u,v)|}{|A|} \geq r \tag{1}$$

where

$$IND(u,v) = \{a \in A : a(u) = a(v)\}, \tag{2}$$

The parameter r is the *granulation radius* from the set $\{0, \frac{1}{|A|}, \frac{2}{|A|}, ..., 1\}$.

2.1 The Process of Training System Covering

In the process of covering - the obejcts of training system are covered based on chosen strategy. We use simple random choice because it is the most effective method among studied ones - see [17]).

The last step of the granulation process is a shown in the next section.

2.2 Granular Reflections

In this step we formed the granular reflections of the original training system based on the granules from the found coverage. Each granule $g \in COV(U, \mu, r)$ from the coverage is finally represented by single object formed using the Majority Voting (MV) strategy.

$$\{MV(\{a(u) : u \in g\}) : a \in A \cup \{d\}\} \tag{3}$$

The granular reflection of the decision system $D = (U, A, d)$ is the decision system $(COV(U, \mu, r)$, the set of objects formed from granules.

$$v \in g_r^{cd}(u) \text{ if and only if } \mu(v, u, r) \text{ and } (d(u) = d(v)) \tag{4}$$

for a given rough (weak) inclusion μ.

Detailed information about our new method of granulation is presented in the next section.

3 Homogenous Granulation

The granules are formed as follows,

$$g_{r_u}^{homogenous} = \{v \in U : |g_{r_u}^{cd}| - |g_{r_u}| == 0,\ for\ minimal\ r_u\ fulfills\ the\ equation\}$$

where

$$g_{r_u}^{cd} = \{v \in U : \frac{IND(u,v)}{|A|} \leq r_u\ AND\ d(u) == d(v)\}$$

and

$$g_{r_u} = \{v \in U : \frac{IND(u,v)}{|A|} \leq r_u\}$$

$$r_u = \{\frac{0}{|A|}, \frac{1}{|A|}, ..., \frac{|A|}{|A|}\}$$

3.1 Toy Example of Homogenous Granulation

Considering training decision system from Table 1.
Homogenous granules for all training objects:
$g_{0.385}(u_1) = (u_1, u_6, u_{10}, u_{11}, u_{12}, u_{18}, u_{20})$,
$g_{0.462}(u_2) = (u_2, u_3, u_4, u_5, u_9, u_{23})$,
$g_{0.539}(u_3) = (u_2, u_3, u_5)$,
$g_{0.615}(u_4) = (u_4)$,
$g_{0.539}(u_5) = (u_3, u_5, u_{21}, u_{23})$,
$g_{0.462}(u_6) = (u_4, u_6, u_{16}, u_{20}, u_{21})$,
$g_{0.539}(u_7) = (u_7, u_{15}, u_{17})$,
$g_{0.462}(u_8) = (u_7, u_8, u_{13})$,

Table 1. Training data system (U_{trn}, A, d), (a sample from heart disease data set)

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}	d
u_1	74.0	0.0	2.0	120.0	269.0	0.0	2.0	121.0	1.0	0.2	1.0	1.0	3.0	1
u_2	65.0	1.0	4.0	120.0	177.0	0.0	0.0	140.0	0.0	0.4	1.0	0.0	7.0	1
u_3	59.0	1.0	4.0	135.0	234.0	0.0	0.0	161.0	0.0	0.5	2.0	0.0	7.0	1
u_4	53.0	1.0	4.0	142.0	226.0	0.0	2.0	111.0	1.0	0.0	1.0	0.0	7.0	1
u_5	43.0	1.0	4.0	115.0	303.0	0.0	0.0	181.0	0.0	1.2	2.0	0.0	3.0	1
u_6	46.0	0.0	4.0	138.0	243.0	0.0	2.0	152.0	1.0	0.0	2.0	0.0	3.0	1
u_7	60.0	1.0	4.0	140.0	293.0	0.0	2.0	170.0	0.0	1.2	2.0	2.0	7.0	2
u_8	63.0	0.0	4.0	150.0	407.0	0.0	2.0	154.0	0.0	4.0	2.0	3.0	7.0	2
u_9	40.0	1.0	1.0	140.0	199.0	0.0	0.0	178.0	1.0	1.4	1.0	0.0	7.0	1
u_{10}	48.0	1.0	2.0	130.0	245.0	0.0	2.0	180.0	0.0	0.2	2.0	0.0	3.0	1
u_{11}	54.0	0.0	2.0	132.0	288.0	1.0	2.0	159.0	1.0	0.0	1.0	1.0	3.0	1
u_{12}	71.0	0.0	3.0	110.0	265.0	1.0	2.0	130.0	0.0	0.0	1.0	1.0	3.0	1
u_{13}	70.0	1.0	4.0	130.0	322.0	0.0	2.0	109.0	0.0	2.4	2.0	3.0	3.0	2
u_{14}	56.0	1.0	3.0	130.0	256.0	1.0	2.0	142.0	1.0	0.6	2.0	1.0	6.0	2
u_{15}	59.0	1.0	4.0	110.0	239.0	0.0	2.0	142.0	1.0	1.2	2.0	1.0	7.0	2
u_{16}	64.0	1.0	1.0	110.0	211.0	0.0	2.0	144.0	1.0	1.8	2.0	0.0	3.0	1
u_{17}	67.0	1.0	4.0	120.0	229.0	0.0	2.0	129.0	1.0	2.6	2.0	2.0	7.0	2
u_{18}	51.0	0.0	3.0	120.0	295.0	0.0	2.0	157.0	0.0	0.6	1.0	0.0	3.0	1
u_{19}	64.0	1.0	4.0	128.0	263.0	0.0	0.0	105.0	1.0	0.2	2.0	1.0	7.0	1
u_{20}	57.0	0.0	4.0	128.0	303.0	0.0	2.0	159.0	0.0	0.0	1.0	1.0	3.0	1
u_{21}	71.0	0.0	4.0	112.0	149.0	0.0	0.0	125.0	0.0	1.6	2.0	0.0	3.0	1
u_{22}	53.0	1.0	4.0	140.0	203.0	1.0	2.0	155.0	1.0	3.1	3.0	0.0	7.0	2
u_{23}	47.0	1.0	4.0	112.0	204.0	0.0	0.0	143.0	0.0	0.1	1.0	0.0	3.0	1
u_{24}	58.0	1.0	3.0	112.0	230.0	0.0	2.0	165.0	0.0	2.5	2.0	1.0	7.0	2

$g_{0.462}(u_9) = (u_2, u_4, u_9)$,
$g_{0.615}(u_{10}) = (u_{10})$,
$g_{0.385}(u_{11}) = (u_1, u_6, u_{11}, u_{12}, u_{20})$,
$g_{0.385}(u_{12}) = (u_1, u_{11}, u_{12}, u_{18}, u_{20})$,
$g_{0.615}(u_{13}) = (u_{13})$,
$g_{0.385}(u_{14}) = (u_{14}, u_{15}, u_{24})$,
$g_{0.615}(u_{15}) = (u_{15})$,
$g_{0.539}(u_{16}) = (u_{16})$,
$g_{0.539}(u_{17}) = (u_7, u_{15}, u_{17})$,
$g_{0.389}(u_{18}) = (u_1, u_2, u_6, u_{10}, u_{12}, u_{18}, u_{20}, u_{21}, u_{23})$,
$g_{0.615}(u_{19}) = (u_{19})$,
$g_{0.462}(u_{20}) = (u_1, u_6, u_{11}, u_{12}, u_{18}, u_{20})$,
$g_{0.462}(u_{21}) = (u_3, u_5, u_6, u_{21}, u_{23})$,

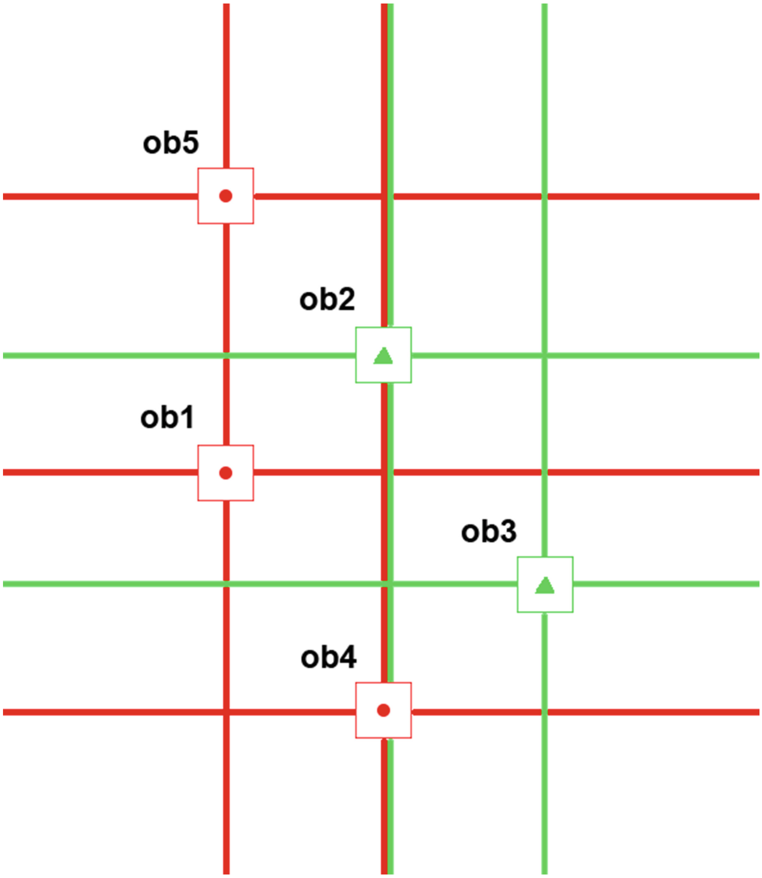


Fig. 1. A simple demonstration of granulation for objects represented by the pairs of attributes. In the picture we have objects of two classes, circles and triangles. Granulating the decision system in homogenous way we can obtain $g_{0.5}(ob1) = \{ob1, ob5\}$, $g_1(ob2) = \{ob2\}$, $g_{0.5}(ob3) = \{ob3\}$, $g_1(ob4) = \{ob4\}$, $g_{0.5}(ob1) = \{ob5, ob1\}$. The set of possible radii is $\{\frac{0}{2}, \frac{1}{2}, \frac{2}{2}\}$

$g_{0.615}(u_{22}) = (u_{22})$,
$g_{0.462}(u_{23}) = (u_2, u_3, u_5, u_{21}, u_{23})$,
$g_{0.462}(u_{24}) = (u_7, u_{15}, u_{24})$,

Granules covering training system by random choice:
$g_{0.462}(u_2) = (u_2, u_3, u_4, u_5, u_9, u_{23})$,
$g_{0.539}(u_3) = (u_2, u_3, u_5)$,
$g_{0.462}(u_6) = (u_4, u_6, u_{16}, u_{20}, u_{21})$,
$g_{0.462}(u_8) = (u_7, u_8, u_{13})$,
$g_{0.385}(u_{12}) = (u_1, u_{11}, u_{12}, u_{18}, u_{20})$,
$g_{0.385}(u_{14}) = (u_{14}, u_{15}, u_{24})$,

$g_{0.539}(u_{17}) = (u_7, u_{15}, u_{17})$,
$g_{0.385}(u_{18}) = (u_1, u_2, u_6, u_{10}, u_{12}, u_{18}, u_{20}, u_{21}, u_{23})$,
$g_{0.615}(u_{19}) = (u_{19})$,
$g_{0.462}(u_{21}) = (u_3, u_5, u_6, u_{21}, u_{23})$,
$g_{0.615}(u_{22}) = (u_{22})$,

Granular decision system from above granules is as follows:

Exemplary visualization of granulation process is presented in Fig. 1.

4 Description Classifier Used for Evaluation of Granulation

In the experiments we use kNN classifier to verify the effectiveness of approximation. The procedure is as follows.

Step 1. The training granular decision system $(G^{trn}_{r_{gran}}, A, d)$ and the test decision system (U_{tst}, A, d) have been input, where A is a set of conditional attributes, d the decision attribute, and, r_{gran} a granulation radius.

Step 2. Classification of test objects by means of granules of training objects is performed as follows.

For all conditional attributes $a \in A$, training objects $v \in G^{trn}$, and test objects $u \in U_{tst}$, we compute weights $w(u, v)$ based on the Hamming metric.
In the voting procedure of the kNN classifier, we use optimal k estimated by CV5, details of the procedure are highlighted in next section.

If the cardinality of the smallest training decision class is less than k, we apply the value for $k = |\textit{the smallest training decision class}|$.

The test object u is classified by means of weights computed for all training objects v. Weights are sorted in increasing order as,

$$w_1^{c_1}(u, v_1^{c_1}) \leq w_2^{c_1}(u, v_2^{c_1}) \leq \ldots \leq w_{|C_1|}^{c_1}(u, v_{|C_1|}^{c_1});$$

$$w_1^{c_2}(u, v_1^{c_2}) \leq w_2^{c_2}(u, v_2^{c_2}) \leq \ldots \leq w_{|C_2|}^{c_2}(u, v_{|C_2|}^{c_2});$$

$$\ldots$$

$$w_1^{c_m}(u, v_1^{c_m}) \leq w_2^{c_m}(u, v_2^{c_m}) \leq \ldots \leq w_{|C_m|}^{c_m}(u, v_{|C_m|}^{c_m}),$$

where $C_1, C_2, ..., C_m$ are all decision classes in the training set.

Based on computed and sorted weights, training decision classes vote by means of the following parameter, where c runs over decision classes in the training set,

$$Concept\_weight_c(u) = \sum_{i=1}^{k} w_i^c(u, v_i^c). \tag{5}$$

Table 2. Granular decision system formed from covering granules

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}	d
$g_{0.462}(u_2)$	65.0	1.0	4.0	120.0	177.0	0.0	0.0	140.0	0.0	0.4	1.0	0.0	7.0	1
$g_{0.539}(u_3)$	65.0	1.0	4.0	120.0	177.0	0.0	0.0	140.0	0.0	0.4	2.0	0.0	7.0	1
$g_{0.462}(u_6)$	53.0	0.0	4.0	142.0	226.0	0.0	2.0	111.0	1.0	0.0	2.0	0.0	3.0	1
$g_{0.462}(u_8)$	60.0	1.0	4.0	140.0	293.0	0.0	2.0	170.0	0.0	1.2	2.0	3.0	7.0	2
$g_{0.385}(u_{12})$	74.0	0.0	2.0	120.0	269.0	0.0	2.0	159.0	0.0	0.0	1.0	1.0	3.0	1
$g_{0.385}(u_{14})$	56.0	1.0	3.0	130.0	256.0	0.0	2.0	142.0	1.0	0.6	2.0	1.0	7.0	2
$g_{0.539}(u_{17})$	60.0	1.0	4.0	140.0	293.0	0.0	2.0	170.0	1.0	1.2	2.0	2.0	7.0	2
$g_{0.385}(u_{18})$	71.0	0.0	4.0	120.0	269.0	0.0	2.0	121.0	0.0	0.0	1.0	0.0	3.0	1
$g_{0.615}(u_{19})$	64.0	1.0	4.0	128.0	263.0	0.0	0.0	105.0	1.0	0.2	2.0	1.0	7.0	1
$g_{0.462}(u_{21})$	59.0	1.0	4.0	112.0	234.0	0.0	0.0	161.0	0.0	0.5	2.0	0.0	3.0	1
$g_{0.615}(u_{22})$	53.0	1.0	4.0	140.0	203.0	1.0	2.0	155.0	1.0	3.1	3.0	0.0	7.0	2

Finally, the test object u is classified into the class c with a minimal value of $Concept\_weight_c(u)$ (Table 2).

After all test objects u are classified, the quality parameter of *accuracy, acc* is computed, according to the formula

$$acc = \frac{number\ of\ correctly\ classified\ objects}{number\ of\ classified\ objects}.$$

4.1 Parameter Estimation in kNN Classifier

In our experiments, we use the classical version of kNN classifier based on the Hamming metric. In the first step, we estimate the optimal k based on 5×CV5 cross–validation on the part of data set. In the next step, we use the estimated value of k in order to find k nearest objects for each decision class and then we vote for decision. If the value of k is larger than the smallest training decision class cardinality then k is mapped on the cardinality of this class.

In Table 3 we can see the estimated values of k for all examined data sets.These values were chosen as optimal on basis of experiments with various values of k and results estimated by means of multiple CV5.

Table 3. Estimated parameters for kNN based on 5×CV5 cross–validation, data from UCI Repository [19]

Name	*Optimal k*
Australian − credit	5
Car Evaluation	8
Diabetes	3
German − credit	18
Heartdisease	19
Hepatitis	3
Nursery	4
SPECTF Heart	14

5 Experimental Session

We have performed five times cross validation test with k-NN classifier on selected data from UCI Repository [19] - see the Table 4. In the experiments we have used k evaluated in our previous works [17]. The list of optimal parameters k is shown in Table 3. Single test consists of splitting the data on training and test set, where the training samples are granulated using our homogenous method. The result of experiments is presented in Table 5. We have shown the comparable effectiveness in comparison with our best concept dependent granulation method - see Eq. 6. The new technique is significantly different from existing methods. Dynamic tuning of radius during granulation results with granules directed on decisions of their central objects. There is no need to estimate optimal radius of granulation, because the radius is selected in automatic way during granulation process. The approximation level depends on objects indiscernibility ratio in the particular decision classes (Table 6).

Table 4. Data Sets description - see [19]

Name	*Attr type*	*Attr no.*	*Obj no.*	*Class no.*
Australian − credit	*categorical, integer, real*	15	690	2
Car Evaluation	*categorical*	7	1728	4
Diabetes	*categorical, integer*	9	768	2
German − credit	*categorical, integer*	21	1000	2
Heartdisease	*categorical, real*	14	270	2
Hepatitis	*categorical, integer, real*	20	155	2
Nursery	*categorical*	9	12960	5
SPECTF Heart	*integer*	45	267	2

Table 5. The result for dynamic granulation - 5 times CV5 method with k-NN classifier; acc_5CV5 = *average accuracy*, GS_size = *granular decision system size*, TRN_size = *training set size*, $TRN\_reduction$ = *reduction in object number in training size*, $radii\_range$ = *spectrum of radii*

Data set	acc_5CV5	GS_size	TRN_size	$TRN\_reduction$	$Radii\_range$
Australian − credit	0.835	286.52	552	48.1%	$r_u \geq 0.5$
Car Evaluation	0.797	728.5	1382	47.3%	$r_u \geq 0.667$
Diabetes	0.653	488.9	614	20.4%	$r_u \geq 0.25$
German − credit	0.725	513.3	800	35.8%	$r_u \geq 0.6$
Heartdisease	0.833	120.5	216	44.2%	$r_u \geq 0.461$
Hepatitis	0.88	46.16	124	62.8%	$r_u \geq 0.579$
Nursery	0.607	9009.1	10368	13.1%	$r_u \geq 0.875$
SPECTF Heart	0.763	138.75	214	35.2%	$r_u \geq 0.068$

Table 6. Exemplary results for our best previously studied approximation method - the concept dependent granulation [17], with use of k-NN classifier, granular and non granular case, *acc* = accuracy of classification, *red* = percentage reduction in object number, r = granulation radius, *method* = variant of Naive Bayes classifier

Name	$k - NN(acc, red, r)$	$k - NN.nil(acc)$
Australian − credit	0.851, 71.86, 0.571	0.855
Car Evaluation	0.865, 73.23, 0.833	0.944
Diabetes	0.616, 74.74, 0.25	0.631
German − credit	0.724, 59.85, 0.65	0.73
Heartdisease	0.83, 67.69, 0.538	0.837
Hepatitis	0.884, 60, 0.632	0.89
Nursery	0.696, 77.09, 0.875	0.578
SPECTF Heart	0.802, 60.3, 0.114	0.779

6 Conclusions

In this paper we have presented preliminary tests for novel homogenous granulation method. The novelty lies in approximation process, which is based on lowering of radius (r-indiscernibility ratio) for central objects until they are surrounded by homogenous objects. The method is forming r-indiscernibility classes in deterministic way and the radii are individually created for all training objects, thus there is no need to search for optimal parameters. We have performed a series of experiments, which shows good effectiveness of the new method in comparison with our best concept dependent algorithm. The reduction of training data size is up to 50 percent of original data with maintenance of internal knowledge in terms of accuracy. In the future works we plan to check the behaviour of new homogenous granulation with numerical data using descriptor indiscernibility ratio and to check the effectiveness of ensemble models. Another direction of research is to find the best classification methods for new approximation algorithm. And finally we are wondering if tolerating a fixed percentage of objects from other classes in the granule can improve the quality of classification.

Acknowledgements. The research has been supported by grant 23.610.007-300 from Ministry of Science and Higher Education of the Republic of Poland.

References

1. Artiemjew, P. : Classifiers from granulated data sets: concept dependent and layered granulation. In: Proceedings RSKD 2007, The Workshops at ECML/PKDD 2007, Warsaw University Press, Warsaw, pp. 1–9 (2007)
2. Artiemjew, P.: A review of the knowledge granulation methods: discrete vs. continuous algorithms. In: Skowron, A., Suraj, Z. (eds.) ISRL, vol. 43, pp. 41–59. Springer, Berlin (2013). https://doi.org/10.1007/978-3-642-30341-8_4
3. Polkowski, L.: Rough Sets. Mathematical Foundations. Physica Verlag, Heidelberg (2002)
4. Polkowski, L.: A rough set paradigm for unifying rough set theory and fuzzy set theory. In: Wang, G., Liu, Q., Yao, Y., Skowron, A. (eds.) RSFDGrC 2003. LNCS (LNAI), vol. 2639, pp. 70–77. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-39205-X_9
5. Polkowski, L.: Formal granular calculi based on rough inclusions. In: Proceedings of IEEE 2005 Conference on Granular Computing GrC05, Beijing, China, pp. 57–62. IEEE Press (2005)
6. Polkowski, L.: A model of granular computing with applications. In: Proceedings of IEEE 2006 Conference on Granular Computing GrC06, Atlanta, USA, pp. 9–16. IEEE Press (2006)
7. Polkowski, L.: The paradigm of granular rough computing. In: Proceedings ICCI 2007, Lake Tahoe NV, pp. 145–163. IEEE Computer Society, Los Alamitos CA (2007)
8. Polkowski, L.: Granulation of knowledge in decision systems: the approach based on rough inclusions. The method and its applications. In: Kryszkiewicz, M., Peters, J.F., Rybinski, H., Skowron, A. (eds.) RSEISP 2007. LNCS (LNAI), vol. 4585, pp. 69–79. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-73451-2_9
9. Polkowski, L.: A unified approach to granulation of knowledge and granular computing based on rough mereology: a survey. In: Pedrycz, W., Skowron, A., Kreinovich, V. (eds.) Handbook of Granular Computing, pp. 375–400. Wiley, Chichester (2008)
10. Polkowski, L.: Approximate Reasoning by Parts: An Introduction to Rough Mereology. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22279-5
11. Polap, D., Wozniak, M., Wei, W., Damasevicius, R.: Multi-threaded Learning Control Mechanism for Neural Networks. Future Generation Computer Systems. Elsevier, Amsterdam (2018)
12. Polkowski, L., Artiemjew, P.: Granular computing: granular classifiers and missing values. In: Proceedings ICCI'07, Lake Tahoe NV, pp. 186–194. IEEE Computer Society, Los Alamitos (2007)

13. Polkowski, L., Artiemjew, P.: On granular rough computing with missing values. In: Kryszkiewicz, M., Peters, J.F., Rybinski, H., Skowron, A. (eds.) RSEISP 2007. LNCS (LNAI), vol. 4585, pp. 271–279. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-73451-2_29
14. Polkowski, L., Artiemjew, P.: Towards granular computing: classifiers induced from granular structures. In: Proceedings RSKD 2007, The Workshops at ECML/PKDD 2007, pp. 43–53. Warsaw University Press, Warsaw (2007)
15. Polkowski, L., Artiemjew, P.: On granular rough computing: factoring classifiers through granulated decision systems. In: Kryszkiewicz, M., Peters, J.F., Rybinski, H., Skowron, A. (eds.) RSEISP 2007. LNCS (LNAI), vol. 4585, pp. 280–289. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-73451-2_30
16. Polkowski, L., Artiemjew, P.: Classifiers based on granular structures from rough inclusions. In: Proceedings of 12th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU 2008, Torremolinos (Malaga), Spain, pp. 1786–1794 (2008)
17. Polkowski, L., Artiemjew, P.: Granular computing in decision approximation - an application of rough mereology. In: Intelligent Systems Reference Library, vol. 77, pp. 1–422. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-319-12880-1. ISBN 978-3-319-12879-5
18. Ohno-Machado, L.: Cross-validation and bootstrap ensembles, bagging, boosting. Harvard-MIT Division of Health Sciences and Technology. HST.951J: Medical Decision Support, Fall (2005). http://ocw.mit.edu/courses/health-sciences-and-technology/hst-951j-medical-decision-support-fall-2005/lecture-notes/hst951_6.pdf
19. University of California, Irvine Machine Learning Repository: https://archive.ics.uci.edu/ml/index.php

Detection of Dental Filling Using Pixels Color Recognition

Oleksandra Osadcha[1](✉), Agata Trzcionka[2], Katarzyna Pachońska[2], and Marek Pachoński[3]

[1] Institute of Mathematics, Silesian University of Technology, Kaszubska 23, 44-100 Gliwice, Poland
osadcha.aleksandra@gmail.com

[2] Department of Conservative Dentistry with Endodontics of Silesian Medical University, Head of Department Prof. Dr. Hab.N.Med. Marta Tanasiewicz, pl. Akademicki 17, 41-902 Bytom, Poland
{atrzcionka,kpachonska}@sum.edu.pl

[3] Private Dental Practice NZOZ “Pachońscy Dental Clinic”, ul. Lubliniecka 38, 42-288 Strzebiń, Poland

Abstract. Dental filling is very important material used to fill the cavity in the tooth, formed after the treatment of caries or as a result of mechanical or other damage to the tooth in stomatology. In this article we show that dental filling can be detected using pixels colors of tooth image to evaluate the size and filling gap. We present an algorithm, which analyzes the size of dental filling and gap of filling. Presented research results show that the developed method can find differences between various types of teeth. Also we use Student t-test for dependent variables, which helps to decide whether there is a difference between different types of teeth.

Keywords: Pixels color detection · Dental filling · Gap of filling Student t-test

1 Introduction

Failures occurring in conservative treatment of caries by the usage of light-cured resin composites are mainly caused by polymerization shrinkage. Each type of a synthetic composite resin (plastic) composed of methacrylate resins in vulnerable to the polymerization contraction that is caused by:

- Decreasing of the distance between particles of monomers in polymer to more or less 1/3 of Van der Waals radius (1,34 Å). Prepolymerization distance between particles containing carbon-carbon bonds (C-C) due to Van der Waals forces is 3 to 4 Å.
- Free volume while transforming of monomers into cross lined polymer.

The polymerization contraction can take the value of even 20% and strongly depends on the kind of monomer and the composition of polymerizing mixtures. It can be a cause of: change of dimensions, forming the empty spaces defects, cracks, bed adhesion of composite to the tooth surface, delamination, crystals’ symmetry deformation. In short

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 347–356, 2018.
https://doi.org/10.1007/978-3-319-99972-2_28

time it can lead to marginal gap formation and as a consequence to the part of the tooth being broken. During the polymerization contraction, the polymerization tension appears. It is responsible for the specific clinical failures, The nano gap that is formed at the beginning enlarges and as a consequence the marginal gap between the filling material and tooth tissues forms. That gap becomes an entrance place for bacteria that cause secondary caries and makes the risk of the tooth crack higher. What influences on the marginal gap formation is also a type of the used adhesive system. If the adhesive force of the used adhesive system is stronger than the force of the shrinkage tension, the marginal gap should not form at all and the shrinkage must be then offset for instance by the flexibility of the material being polymerized. However, described mechanism of compensation does not always take place, all the more that the contraction tension may maintain for a long time. The maintaining contraction tension is a reason of weakening the bonding of material to the tooth stricter and of the gap formation.

In order to decrease the risk of the described phenomenon occurrence, the producers keep modifying both the composites and the adhesive systems. At the same time the dentists are recommended to fill the cavities with the resin composites by the usage of different filling techniques:

- Layering technique utilizes composite resin layers, each <2 mm thick, polymerized separately. That technique does not eliminate the occurrence of the contraction in the whole body of the filling and what is more there is a risk that the bonding between layers of composites would not be complete.
- Double density technique, in which it is recommended to use the materials of low density to fill the bottom of the cavity (2 or 3 layers, each polymerized separately) and then the conventional composites. There is the same risk as in the previous technique- both the occurrence of the contraction is not completely eliminated and the bonding of the layers might not be complete.
- Sandwich technique. In that technique the glass-ionomer is used to replace the dentin and then the composite resin is used. The contraction occurs only in the layer of composite.
- The usage of both chemo-mechanical and light-cured composites. First type of composites is used to replace the dentin.
- The usage of inserts. Small portions of the spherical shaped composites polymerized outside the oral cavity are placed to the portion of not polymerized composite located in the cavity. Even though the contraction still occur it concerns volumetrically small amount of the composite.
- The usage of polymerization lamps with variable power of the light- it does not influence on the value of the polymerization contraction but causes more favorable dispersion of tensions that appear during polymerization which proceeds more orderly.
- Indirect technique. The cavity is completely filled with the composite polymerized outside the oral cavity.

Nowadays composites are commonly used. That is why there is a need to look for the new solutions of minimalizing the risk of polymerization shrinkage occurrence.

1.1 Related Works

There are many different works which present detection of pixels color or special features encoded in key-points. Some tests of color image segmentation are presented in [1]. Also we have digital image that includes first and second regions in the processed object. An intrinsic color of a given pixel located in an area of interest that is adjacent to at least one of the first and second regions is estimated by extrapolating from colors of multiple pixels in one of the first and second regions and multiple pixels in the other of the two regions [2]. In another works a categorizing of pixel colors into basic color categories was discussed to simplify the methodology [3]. In [4] was presented a method, which defines saliency values for image pixels using color statistics of the input image. Authors of [5] proposed a technique to select key features from color objects, while in [6] was proposed a methodology to define bacteria from medical images using their shapes represented in fluorescent screenings. On the other hand an important part of medical examinations is based on x-ray or computed tomography. In [7] this type of images were used for evaluation of dental bones problems, while in [8] was presented a review of computer techniques usable in kidney examinations from medical images. Other important aspect of using medical images in examinations is lung disease. In [9] was proposed to use key-points extraction for automated decision support, while in [10, 11] lung nodule and lymph node were evaluated by the use of computer methods.

In this article we present an automated technique developed for estimation of dental fillings. In this method we have developed an algorithm which is used to scan an input dental image to search for the tooth filling. As an output our method returns an image with marked regions of the tooth filling and, what is even more important, the exact border of this filling. Our method can be used for evaluation of the fillings, their medical accuracy and potential efficiency after tooth surgery at the dentist.

2 How Do We Detect Dental Filling?

The proposed algorithm has been implemented in C# Microsoft Visual Studio 2013.

At first, we load image of tooth and rewrite image in pixels form, also we find size of image. To facilitate the search of the necessary pixels of the dental filling, first, we convert the input color image to a grayscale. After that we present the image in RGB color model, then we check how the RGB *values* of the dental filling pixels change. We search on each coordinates for possible locations of the dental filling changes and count the number of these pixels. For example, if *values* changes in $[x, y]$ range, in our program we rewrite it in the following condition: *if (values > = x && values <= y)*. To check availability of RGB *values* in the given position, we choose the range of image coordinates. For example, this condition is next: *(150 < i && i < 350) && (j > 600 && j < 800),* where *(i, j)* – coordinates of the image, (150, 600) and (350, 800) is range of coordinates. We use RGB color model, because in Visual Studio it is very easy to convert the image to RBG model. First, we create Bitmap (it consists of the pixel data for a graphics image and its attributes) variable and three two-dimensional arrays. In these we store coordinates values of each color. Next, we create *Color* structure, which represents R, G, B colors and write coordinates of these colors into arrays, for example the code is

```
Color c = bmp.GetPixel(i, j);
Color_red[i, j] = Colors[i, j].R;
Color_green[i, j] = Colors[i, j].G;
Color_blue[i, j] = Colors[i, j].B;
```

In these way, we store R, G, B values. After counting, we analyze how many percent of the filling takes the whole tooth to present the result to the doctor.

In the next section we discuss the statistics for each tooth. We can see the quantity of whole tooth filling pixels, it's filling and the gaps.

3 Experiments

In the research we have tested proposed model on sample images presenting various objects. Pictures were taken by the usage of Dental Microscope Zumax, with magnification of 20x. We have analyzed 18 examples of teeth (Figs. 1, 2, 3, 4, 5 and 6).

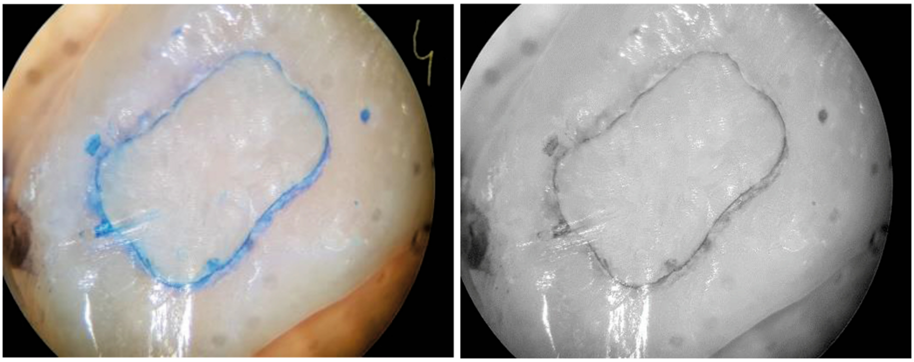


Fig. 1. Color and grayscale photos of the tooth 1. (Color figure online)

Nine of them polished immediately and the others polished after one day. The main aim of these tests was to compare which type of the tooth has smaller gap of the filling. The red selection represents how well the filling adheres to the tooth specially the wall (we called it gap). The green selection shows the filling of the whole tooth. In Table 1 is presented statistic of each tooth, where we can see quantity of the tooth, the filling and the gap but also the coverage presented in percentage measure in relation to the whole object.

We prepared detailed statistical analysis, by Student's t-test, Wilcoxon and Sign tests.

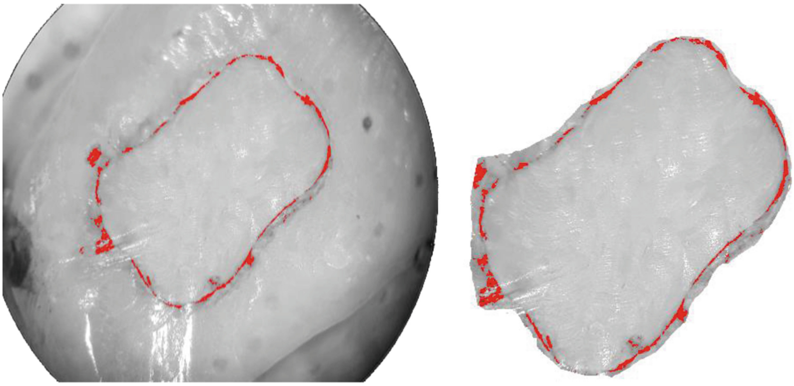


Fig. 2. Filling gap of tooth 1.

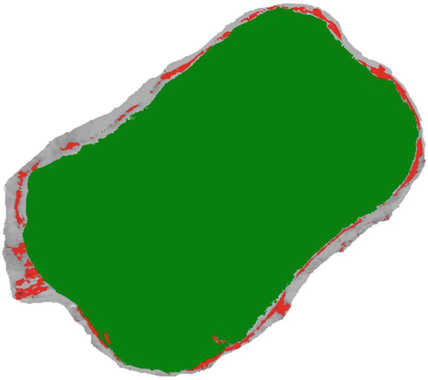


Fig. 3. Dental filling and the gap of tooth 1.

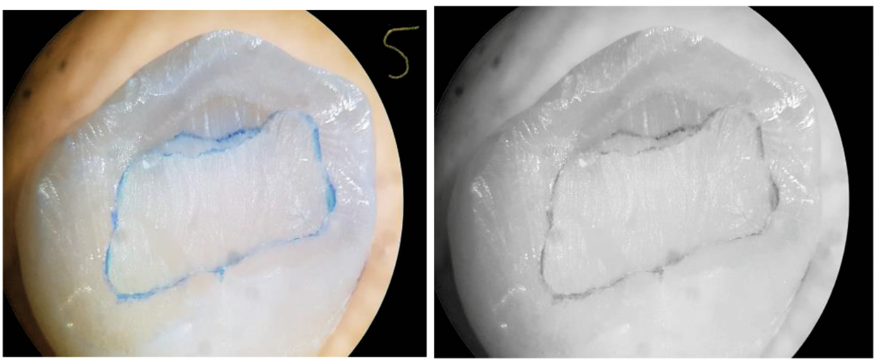


Fig. 4. Color and grayscale photos of tooth 2. (Color figure online)

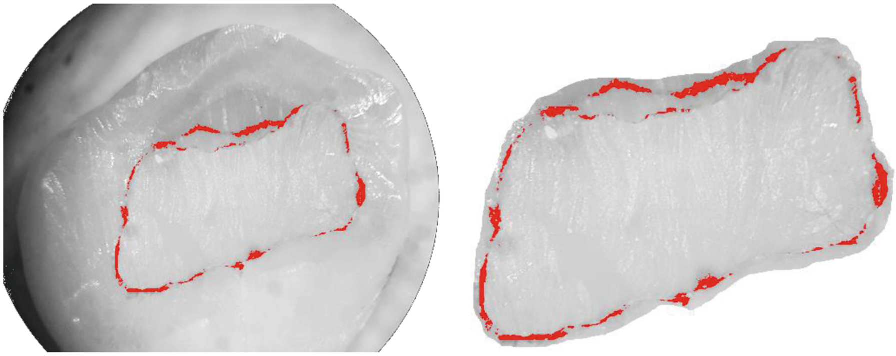


Fig. 5. Filling gap of tooth 2.

Fig. 6. Dental filling and the gap of tooth 2.

In program "Statistica" we have use Student's t-test for dependent variables to check if the difference between these teeth are significant or not. We adopted this level of significance $\alpha = 0,05$ (Fig. 7).

Variable	Student's t-test for dependent variables The differences are important with p < 0,05				
	Average	Stand.dev.	Difference	Stand.dev. Difference	t
immediately	58277,56	60208,40			
after 1 day	71424,33	50856,79	-13146,8	90079,27	-0,437840

Fig. 7. Results of Student's t-test in "Statistica".

When we compare α and p, we can see, that $p < \alpha$. It means, that there is significant difference between these types of teeth.

Variables	Wilcoxon signed-rank test The differences are important with p <0,05	
	T	Z
immediately & after 1 day	11,00000	0,507093

Fig. 8. Results of Wilcoxon test in "Statistica".

Variables	Sign test The differences are important with p <0,05	
	Percentage v < V	Z
immediately & after 1 day	71,42857	0,755929

Fig. 9. Results of Sign test in "Statistica".

The Wilcoxon and Sign tests also show, that $p < \alpha$ and these types of teeth are different (Figs. 8 and 9).

The Student t-test showed that differences are significant, because $p < 0,05$ and $p < \alpha$. In Fig. 10 we can see difference between both measured examinations.

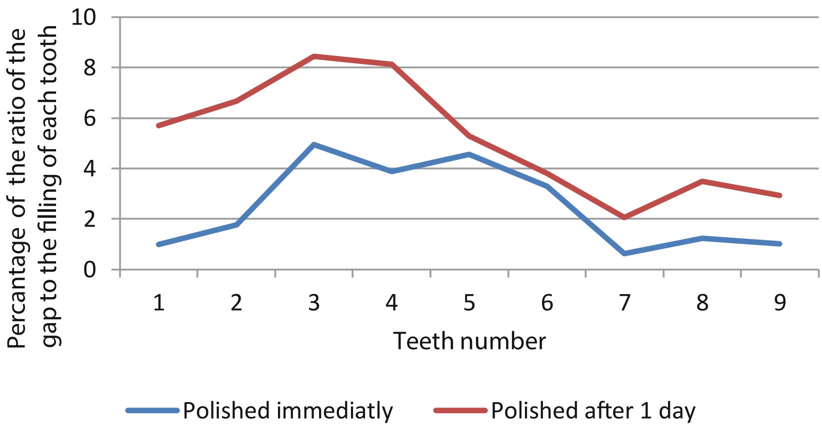


Fig. 10. The ratio of the gap to the filling, which presents the difference between two types of teeth, polished immediately after surgery and polished 1 day after the surgery.

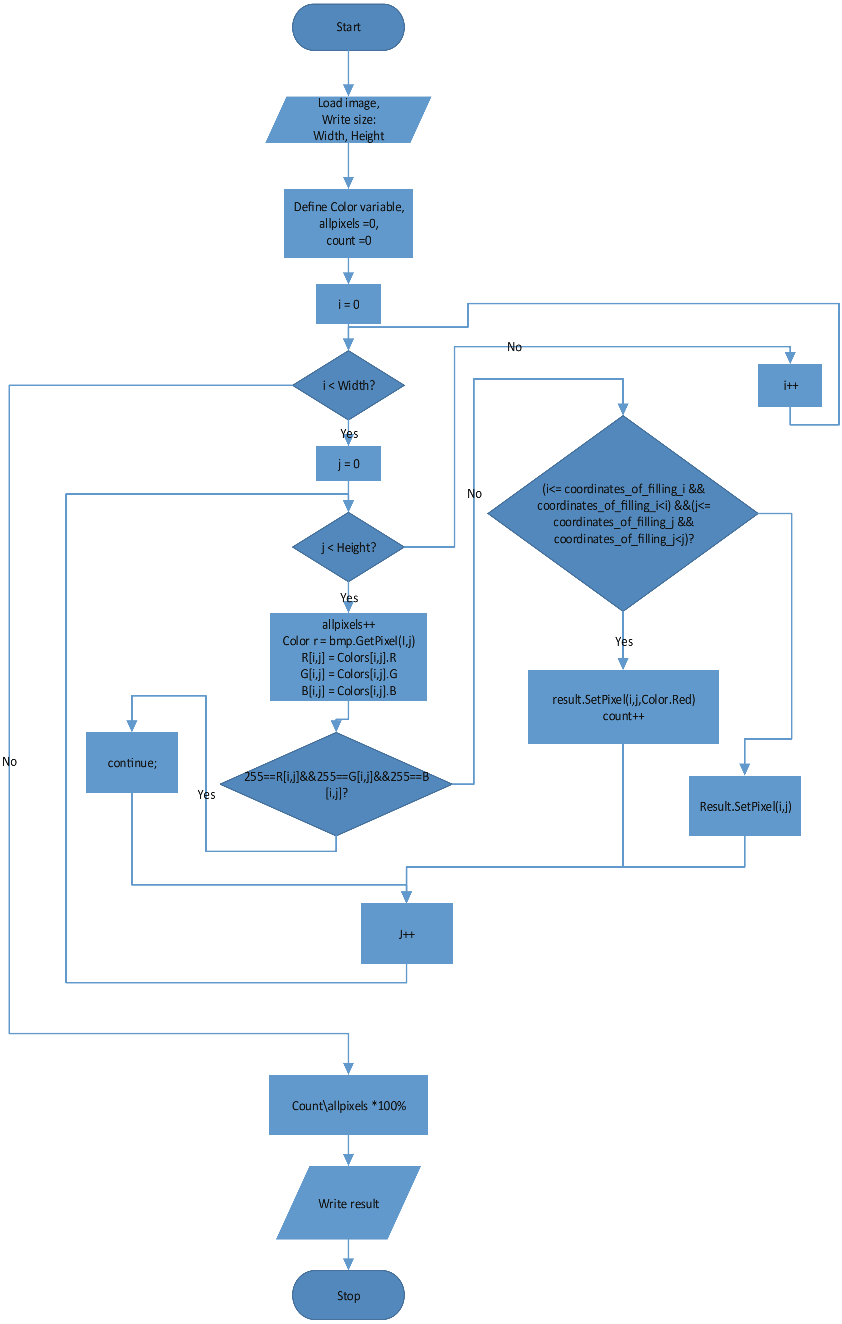


Fig. 11. Block scheme diagram of pixel algorithm recognition

After comparing our results we can see that teeth, which were polished immediately have smaller gap and that there are significant differences between these types of teeth, we can see it in Table 1. The results of the Student t-test show that these types are different.

Table 1. Statistics in percentage (%) of 18 teeth, where one teeth were polished immediately after surgery, another were polished 1 day after the surgery.

Teeth statistic in percentage (%)			
Number of tooth	The ratio of the gap to whole tooth	The ratio of the filling to the whole tooth	The ratio of the gap to the filling
Teeth, which polished immediately after surgery			
1	0,4091	41,8376	0,97783
2	0,4316	24,395	1,76927
3	0,1912	19,1184	4,935
4	0,840586767	21,625707	3,88697935
9	1,6648	36,5137	4,55948
10	0,094196	28,50235	3,30488
11	0,376775	42,96878	0,63067
12	0,50025	40,95452	1,22148
18	0,218658	21,54832	1,01473
Teeth, which polished after 1 day after surgery			
5	1,03934167	22,0328452	4,71723765
6	1,44375	29,3994	4,91081
7	1,27341	36,35302	3,5029
8	12,37258	29,255	4,2307
13	0,26794	36,4043	0,73602
14	0,17631	35,74398	0,49327
15	0,62976	43,98726	1,43169
16	1,10762	48,675231	2,275536
17	0,6267995	32,76589	1,912963

4 Conclusions

Research results have shown that our method can be used for automated finding of the dental filling and the gap of the filling. Proposed technique for pixels recognition helped us to verify the size of the filling and the filling gap. After verification it is easy to conclude, which type of teeth has smaller or bigger gap.

5 Final Remarks

This method shows that pixel verification can be used not only in informatics, but also in medicine. Proposed method is based on recognition of pixel colors. This detection is done using range of RGB values. Each pixel has RGB values, so it easy to find the color of pixels. The novelty of the proposed method is that we can use it in medicine for detection of dental filling size and it gap. The results of the research show that proposed method works properly showing differences between teeth, which were polished immediately and teeth, which polished after 1 day. The result of the research is that the teeth, which polished immediately after surgery have smaller gap. It means, that dental filling is closer to the wall of the tooth, so filling will keep better.

From the dentist's point of view, the problem of polymerization shrinkage is the biggest worry of contemporary dentists. The gap between the composite and tooth tissue is not visible immediately after the filling is put. Dentists are able to observe that after some time when the gap is filled with the dyes coming from food or if there appears secondary caries.

In future work, we plan to improve the model of search for more complex cases and other variants of the filling.

Acknowledgments. Authors acknowledge contribution to this project of the Silesian University of Technology and Silesian Medical University.

References

1. Deng, Y., Manjunath, B.S., Shin, H.: Color image segmentation. Department of Electrical and Computer Engineering University of California, Santa Barbara, CA 93 106-9560 Samsung Electronics Inc. https://doi.org/10.1109/cvpr.1999.784719
2. Wilensky, G.D., Newell, M.E.: Identifying intrinsic pixel colors in a region of uncertain pixels. US 2004/0042662 A1, 4 March 2004
3. Chang, Y., Saito, S., Nakajima, M.: A framework for transfer colors based on the basic color categories. https://doi.org/10.1109/cgi.2003.1214463
4. Cheng, M.-M., Zhang, G.-X., Mitra, N.J., Huang, X., Hu, S.-M.: Global contrast based salient region detection. https://doi.org/10.1109/tpami.2014.2345401
5. Woźniak, M., Połap, D.: Bio-inspired methods modeled for respiratory disease detection from medical images. In: Swarm and Evolutionary Computation (2018)
6. Woźniak, M., Połap, D.: Adaptive neuro-heuristic hybrid model for fruit peel defects detection. Neural Netw. **98**, 16–33 (2018)
7. Woźniak, M., Połap, D., Kośmider, L., Cłapa, T.: Automated fluorescence microscopy image analysis of Pseudomonas aeruginosa bacteria in alive and dead stadium. Eng. Appl. Artif. Intell. **67**, 100–110 (2018)
8. Gan, Y., Xia, Z., Xiong, J., Li, G., Zhao, Q.: Tooth and alveolar bone segmentation from dental computed tomography images. IEEE J. Biomed. Health Inf. **22**(1), 196–204 (2018)
9. Torres, H.R., Queirós, S., Morais, P., Oliveira, B., Fonseca, J.C., Vilaça, J.L.: Kidney Segmentation in Ultrasound, Magnetic Resonance and Computed Tomography Images: A Systematic Review. In: Computer Methods and Programs in Biomedicine (2018)
10. Teramoto, A., Fujita, H.: Automated lung nodule detection using positron emission tomography/computed tomography. In: Suzuki, K., Chen, Y. (eds.) Artificial Intelligence in Decision Support Systems for Diagnosis in Medical Imaging, pp. 87–110. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-68843-5_4
11. Cooper, L.J., Zeller-Plumhoff, B., Clough, G.F., Ganapathisubramani, B., Roose, T.: Using high resolution x-ray computed tomography to create an image based model of a lymph node. J. Theor. Biol. **449**, 73–82 (2018)

The Use of an Artificial Neural Network for a Sea Bottom Modelling

Jacek Lubczonek and Marta Wlodarczyk-Sielicka(✉)

Institute of Geoinformatics, Faculty of Navigation,
Maritime University of Szczecin, Szczecin, Poland
{j.lubczonek,m.wlodarczyk}@am.szczecin.pl

Abstract. Currently data are often acquired by using various remote sensing sensors and systems, which produce big data sets. One of important product are digital models of geographical surfaces that include the sea bottom surface. To improve their processing, visualization and management is often necessary reduction of data points. Paper presents research regarding the application of neural networks for bathymetric geodata reductions. Research take into consideration radial networks, single layer perceptron and self-organizing Kohonen network. During reconstructions of sea bottom model, results shows that neural network with less number of hidden neurons can replace original data set. While the Kohonen network can be used for clustering during reduction of big geodata. Practical implementation of neural network with creation of surface models and reduction of bathymetric data is presented.

Keywords: Bathymetric data · Neural networks · Interpolation
Reduction · DTM · Big data

1 Introduction

Surface modelling was previously an objects of various researches [5, 10], but more focused on numerical methods. Mainly in surface reconstruction process were used some mathematical formulas and sample data set. Created surfaces as raster model are currently used for many tasks, especially in Geographic Information Systems, which are based on digital geographical surface processing [11]. Also for this purpose can be applied neural networks, which are widely used in many applications. Part of this application can be found in marine application, where they were used for ship positioning [13], object tracking [12] or bottom model building [8, 14]. Presently for solving various tasks can be used various type of neural networks, e.g. radial networks, self-organising networks or multilayer perceptron. One of property of neural network is function approximation of many variables, what can be adopted for surface modelling. But now data are collected in the sets of even million points by using the new remote sensing technology like LIDAR, photogrammetry, radar interferometry or multibeam echosounder. Even for numerical methods used in surface creation such big data sets are problems in their processing. It is obvious that this problem could be solved by data reduction, where some minimum number of points would be enough for surface reconstruction within limits of tolerance. By application of neural networks, this

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 357–369, 2018.
https://doi.org/10.1007/978-3-319-99972-2_29

problem could be solved in another way. By proper adjusting their structure can be reduced number of their elements, like number of layers or neurons. In that way smaller set of data which represents neural networks structure could replace the bigger data set of measurement points. Another approach may consist in the physical reduction of the number of measuring points. This paper shows possibility of use radial network, multilayer perceptron and the self-organising networks for above aims.

The paper is ordered as follows: Sect. 2 includes specification of artificial neural network used during the tests and description of test surfaces that reflect the shape of the bottom; Sect. 3 describes methodology of conducted analysis and the experiments results and the publication ends with conclusions.

2 Background

2.1 Artificial Neural Network

In the experiment were used three kinds of neural networks. The first one was radial networks. Properties of radial networks were studied by many researches earlier in the aspect of surface shape modelling, firstly as numerical methods [3, 4] and later as widely applied neural networks. A radial network consists of three layers, one input layer, one hidden layer and one output layer. Multilayer perceptron, beside of one input and output layer, can have many hidden layers. But on the basis of researchers, also one hidden layer is enough for surface modelling purpose [1]. It also simplifies network structure optimisation to estimation number of hidden layers, type of transfer functions, data normalisation methods and training algorithm. For the aim of data reduction, the main role plays number of neurons of hidden layer and weights. Less hidden neuron, less data is required for restoring of network structure. Radial neural network approximation function can be express according following formulae [2, 6]:

$$f(x) = \sum_{i=1}^{K} W_i \varphi_i(\|x - t_i\|), \tag{1}$$

where: φ – radial basis function, K – number of radial basis function, W – networks weights, x – input vector, t – centre of radial basis function.

Having p data points (x, y, z), weights vector W could be calculated by solving system of linear equations:

$$W = H^{+} z \tag{2}$$

$$H^{+} = (H^T H)^{-1} H^T \tag{3}$$

where:

$$W = [\, W_1 \quad W_2 \quad \ldots \quad W_k \,]^T \tag{4}$$

$$H = \begin{bmatrix} \varphi\|x_1 - t_1\| & \varphi\|x_1 - t_2\| & \dots & \varphi\|x_1 - t_k\| \\ \varphi\|x_2 - t_1\| & \varphi\|x_2 - t_2\| & \dots & \varphi\|x_2 - t_k\| \\ \dots & \dots & \dots & \dots \\ \varphi\|x_p - t_1\| & \varphi\|x_p - t_2\| & \dots & \varphi\|x_p - t_k\| \end{bmatrix} \tag{5}$$

$$z = \begin{bmatrix} z_1 & z_2 & \dots & z_k \end{bmatrix}^T \tag{6}$$

When number of hidden neurons is equalled to number of training data, network can be used as interpolation function. By decreasing number of hidden neurons, neural network is an approximator, thus we can expect some losses in surface reconstruction. On the other hand, data loss can be accepted to some tolerance limits. Similar situation is in the case of multilayer perceptron, where number of hidden neurons decides on error of surface approximation. For estimation the influence of hidden neurons on that value in the experiment were used neural networks with structure and training algorithms presented in Table 1.

Table 1. Neural networks structure and training algorithm used for reconstruction of bottom model

	Radial network	SLP (Single Layer Perceptron)
Number of input, hidden, and output layer	1	1
Number of neurons in input layer/transfer function type	2/linear	2/linear
Number of neurons in output layer/transfer function type	1/linear	1/linear
Number of neurons in hidden layer/transfer function type	various/multiquadric	various/sigmoidal
Pre-processing of training set method	Min-Max normalisation	Min-Max normalisation
Training algorithm	k-means with mechanism of neuron fatigue	Quasi-Newton
Training epochs number	4000	4000

An important aspect is the optimization of the radius of the radial function, what was also the subject of researches [7, 15]. For optimization of radial basis function in the RBF hidden layer was applied algorithm, which automates this process. It is based on Cross Validation and Leave-One-Out method [9].

In the case of large datasets of bathymetric points, the Kohonen network may be used during their reduction. Such networks learning there is no relationship between input and the output of the network. The competition between neurons supplies the basis for updating values assigned to their weights. It can be assumed that x is the input vector, p is the number of input samples, w is the weight vector and connected with the node l.

$$x = \left(x_1, x_2, x_3, \ldots, x_p\right)' \tag{7}$$

$$w_l = \left(w_{l1}, w_{l2}, w_{l3}, \ldots, w_{lp}\right)' \tag{8}$$

The individual samples of the training dataset are presented to the Kohonen network in the unsystematic order. In the next step, the nearest neuron to the input sources is a winner for the input dataset. The extent of adaptation depends on the distance of the neuron from the input dataset. The node l is shifted some proportion of the distance between it and the training simple. This proportion is depended on the learning rate. For several objects i, the distance between the weight vector and the input is evaluated. After the start of the competition the node l with the nearest distance is the winner. Next, the weights of the winner are updated using the learning rule (Eq. 10). The weight vector for the lth node in the sth step of training can be described as w_l^s and the input vector for the ith training simple can be presented as X_i. After several epochs, a training simple is selected and the index q of the winning node is defined:

$$q = arg\,\underset{l}{min}\left\|w_l^s - X_i\right\| \tag{9}$$

The Kohonen updated rule is as follows [18]:

$$w_q^{s+1} = w_q^s(1 - \alpha^s) + X_i\alpha^s = w_q^s + \alpha^s\left(X_i - w_q^s\right) \tag{10}$$

where α^s is the learning rate for the sth step of training [16].

2.2 Test Surfaces

Numerical experiment was based on three surface types, which can represent various forms of real surface. Various shapes of surfaces allowed assessing this method for various degree of surface curvature. These surfaces were created by using three test functions, denoted as TF:

$$TF1: \quad f(x, y) = \frac{\tanh(9x - 9y) + 1}{9} + \frac{(\cos(5.4y))^2}{6(1 + (3x - 1)^2)}, \quad x, y \in [0.1, 1] \tag{11}$$

$$TF2: \quad f(x, y) = \frac{\tanh(9x - 9y) + 1}{9}, \quad x, y \in [0.1, 1] \tag{12}$$

$$TF3: \quad f(x, y) = \frac{1.25 + \cos(5.4y)}{6 + 6(3x - 1)^2}, \quad x, y \in [0.1, 1] \tag{13}$$

Finally, after rescaling, domain of surface being modelled can be written as:

$$D = \{(x, y)| \quad 1\ m \leq x \leq 100\ m,\ 1\ m \leq y \leq 100\ m\} \tag{14}$$

Surfaces heights were also rescaled by multiply value $f(x, y)$ by 100. The aim of surface value rescaling was to obtain more convergence with topographical surfaces, which are metric products. Depths of surfaces 1 and 3 change from 5 to 25 m, and for surface 2 from 5 to 15 m. Surfaces, created by using test functions, are presented in Fig. 1.

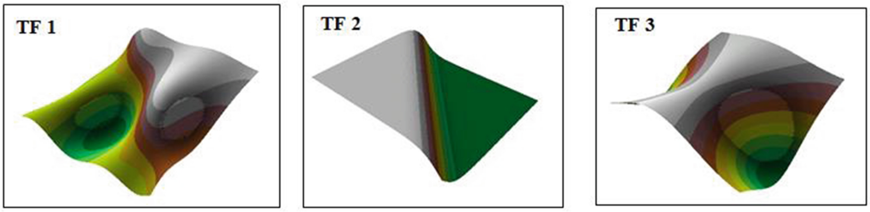


Fig. 1. Shape of test surfaces

In the first part of the experiment four data sets were used. In general, there were two types of spatial distribution of data points – scattered and regular. They additionally also differed in the number of points: $S_1 = \{(x_i, y_i)\colon\ i = 1,2\ldots100\}$, $S_2 = \{(x_i, y_i)\colon\ i = 1,2\ldots200\}$, $S_3 = \{(x_i, y_i)\colon\ i = 1,2\ldots100\}$, $S_4 = \{(x_i, y_i)\colon\ i = 1,2\ldots200\}$. Spatial distribution of data points are presented in Fig. 2.

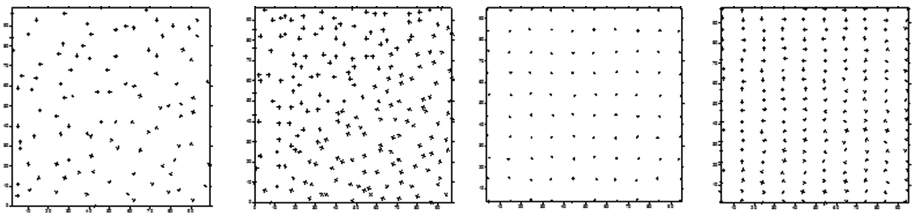


Fig. 2. Spatial distribution of data points in S_1, S_2, S_3, S_4 datasets.

Surface approximation error was computed for coordinates of nodes of regular GRID with resolution of 1 m within domain of surface being modelled according to the formula:

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(f(x_i, y_i) - G(x_i, y_i)\right)^2} \tag{15}$$

where N is the number of points used for error calculation (10000), $f(x_i, y_i)$ is calculated value and $G(x_i, y_i)$ is the real value.

In the last part of the research the S_5 dataset for each test surface was reduced. The spatial distribution of S_5 dataset is scattered and it contains 200 000 points.

3 The Experiment

Experiment was performed on mathematical test functions. The aim of first part of the experiments was to study influence of hidden neuron reduction on surface approximation error. It potentially enables data reduction by replacing the real data by less number of parameter required to remember neural network structure. In the last step of the experiment the use of the self-organizing network for reduction and clustering of spatial data has been presented.

3.1 Interpolation of Bathymetric Data

Experiment with radial networks was conducted for two cases. In the first the number of neurons in hidden layer was equalled number of data in training set. The next case was performed for hidden neurons reductions. The initial value of hidden neurons was equalled number of data points. Results were presented in Fig. 3.

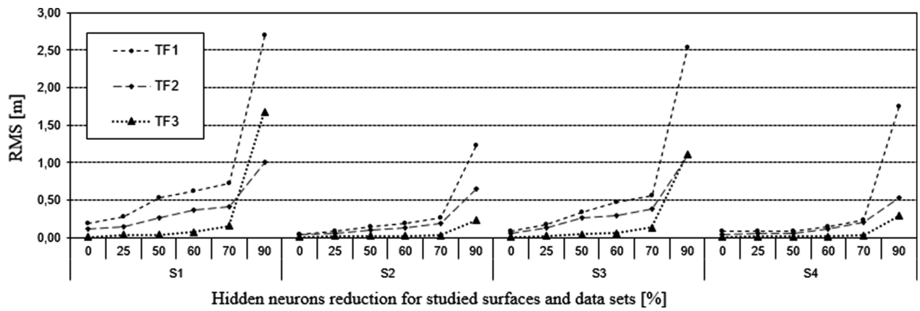


Fig. 3. Influence of neuron reduction in hidden layer on RMS error of research cases for radial neural networks

Analysing above results we can stated that more reduction neurons in hidden layer increase RMS errors values. For exact solution, where number of hidden neurons is the same as number of data points, RMS errors have the smallest values. By increasing neurons reduction, RMS errors increased, but a bit different for each surface type and data set. Biggest values are for TF1 and for data sets with less data density – S_1 and S_3. These results confirms the rule that less data in surface creation process and more irregular surface shape will lead to worse results. Dependence of RMS error on hidden neuron reduction for all studied cases was illustrated in Fig. 4.

As we can note, the values of RMS errors are relatively low for even 60% reduction of hidden neuron (0.22 m). For more surface generalization the tolerance value as a rule is bigger, so can be applied even reduction on the level of 90%.

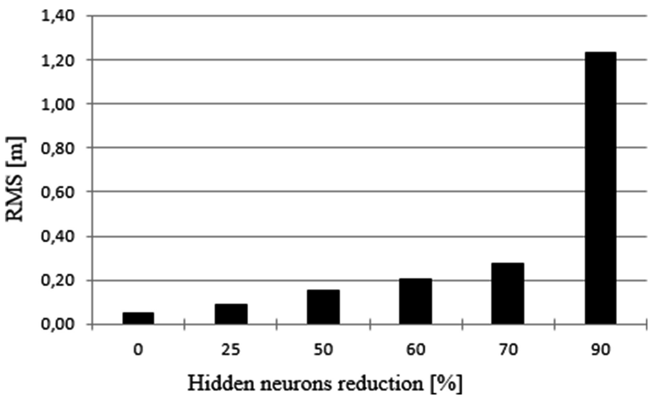


Fig. 4. Dependence of RMS error on hidden neuron reduction.

The same examinations path was performed for multilayer perceptron. The initial value of hidden neurons was equalled 50. Values of RMS errors were presented on Fig. 5. In the case of test function 2 the RMS was equalled almost 0 due to its sigmoidal shape.

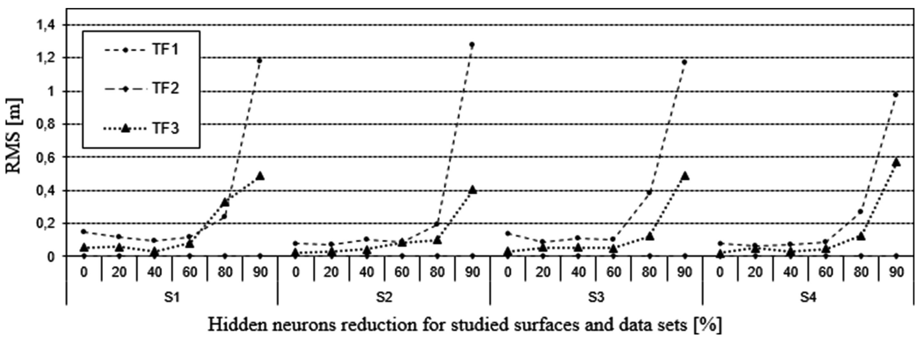


Fig. 5. Influence of neuron reduction in hidden layer on RMS error of research cases for MLP

Comparing this results to neuron reduction in RBF networks, can be noted, that RMS values till 60–80% reduction generally have similar values. Beyond these thresholds they can rapidly increase. Also noticeable is less influence of surface shape and spatial data distribution. Dependence of tolerance RMS error on hidden neuron reduction for all studied cases was illustrated in Fig. 6.

The next difference is almost equaled values in the reduction range of 0–60%. For RBF networks less hidden neuron reduction successive decreases RMS error values. Overall better results can be achieved by using radial networks. The next advantage is that knowing this functional relation is easiest to assess the surface approximation error by analyzing hidden neuron reduction. For SLP, to shorten

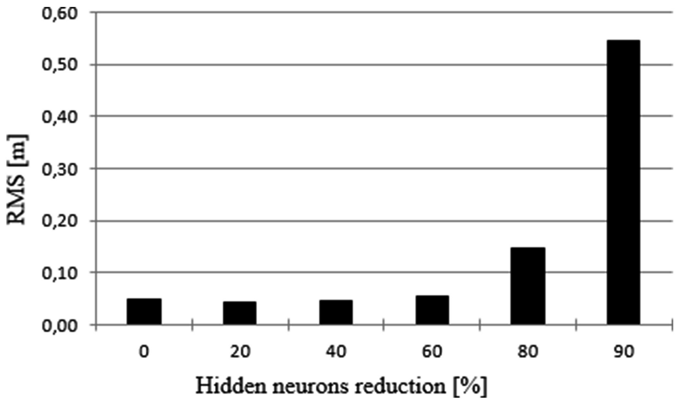


Fig. 6. Dependence of RMS error on hidden neuron reduction.

training process, rational choice is less number of hidden neurons. On the other hands, by increasing their number, RMS error is not significantly improved.

Reduction of hidden neurons allows decreasing data, which are required for surface reconstruction. Comparing digital numbers required to remember neural networks weights and real data set, the data reduction can be quite big (Fig. 7). E.g. for 30 hidden neurons reduction for SLP is 80% and for RBF – 70%.

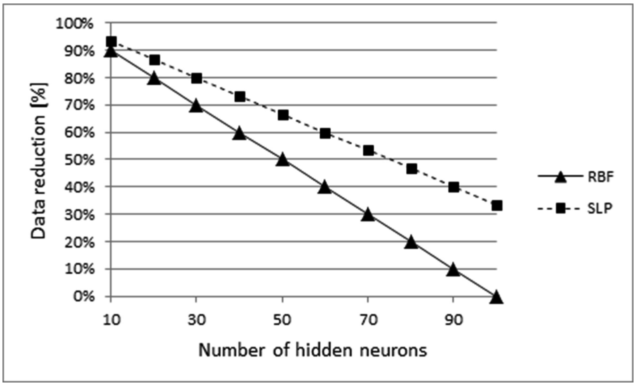


Fig. 7. Assessment of data reduction depending on the number of neurons for the research cases

It should be remembered, however, that in this case the reconstructed surface is approximated and does not retain the real points. The next section describes the use of neural networks for the physical reduction of a data point set using the Kohonen network. This means that this method preserves real measuring points.

3.2 Reduction of Bathymetric Data

The next stage of experiment was to use of artificial neural networks during the reduction of large bathymetric data sets. The research related to the reduction was carried out on the of S_5 dataset.

At this stage the original reduction method was used [18]. The created method of bathymetric geodata reduction is composed of three basic stages. The first stage of the reduction method is initial data preparation for their preprocessing. This stage is composed of two steps: preliminary division and division into grid square. The first step- the preliminary division is necessary in case of large measuring areas and consists of creating the sets of data which would be divided into the grid square. The grid square is based on the quad tree [17]. During the initial processing the bathymetric data are prepared for the next stage, known as data clustering with the usage of the artificial neural network. During the research all the parameters of the self-organizing network were analyzed: neighborhood topology and its initial size, category of distance, number of epochs and the type of the rule used for trainings neurons. The optimal values of parameters were chosen, the usage of which fulfils the assured criteria. The obtained number of clusters would be dependent on both the scale of the map and the outline for the position of each element of the geodata set [19]. Measuring points from the particular clusters will undergo the process of reduction. The most important would be the points of the smallest depth. A circle around each point will be delineate, a size of a circle is strongly dependent on the characteristic of the tested sample. Around the objects of greater importance bigger circles are created- the smaller the depth the bigger the circle. Next step is the reduction of the points located in the circle round the point of the higher importance, so the one of the smaller depth. The process will be repeated until only the circles for the objects of the higher importance are left.

During the second stage of method, based on previous studies, for each trainings the number of epochs at 200 was set. The number of training steps for initial coverage of the input space was set at 100. Network applies the rule Winner Take Most and distances from a particular neuron to its neighbors were calculated as Euclidean distance. The hexagonal topology and initial neighborhood size equal to 10 were used.

The test dataset was reduced for three scales: 1:500, 1:1000 and 1:2000. To evaluate how the created method of bathymetric geodata reduction works, the following criteria were taken into consideration: visual assessment of obtained points' distribution and surfaces, calculation of the percentage decrement of data after reduction (how much is the set of points smaller after reduction) and statistics related to the depth value.

After implementation of the assumed parameters for each scales, we obtained nine datasets after reduction. From all the collections using the Triangulation with Linear Interpolation method (implemented in ArcGIS software), the surfaces were modeled. The surfaces with spatial distribution of bathymetric data points for each scale are presented in Fig. 8.

Visualization of depth points with their description fulfils the assumed criteria of evaluation. The points have been significantly reduced and their distribution is regular. The position and depth values have not been interpolated. Surfaces obtained from the redundant points built by the usage of method in 1:500 scale almost perfectly illustrate

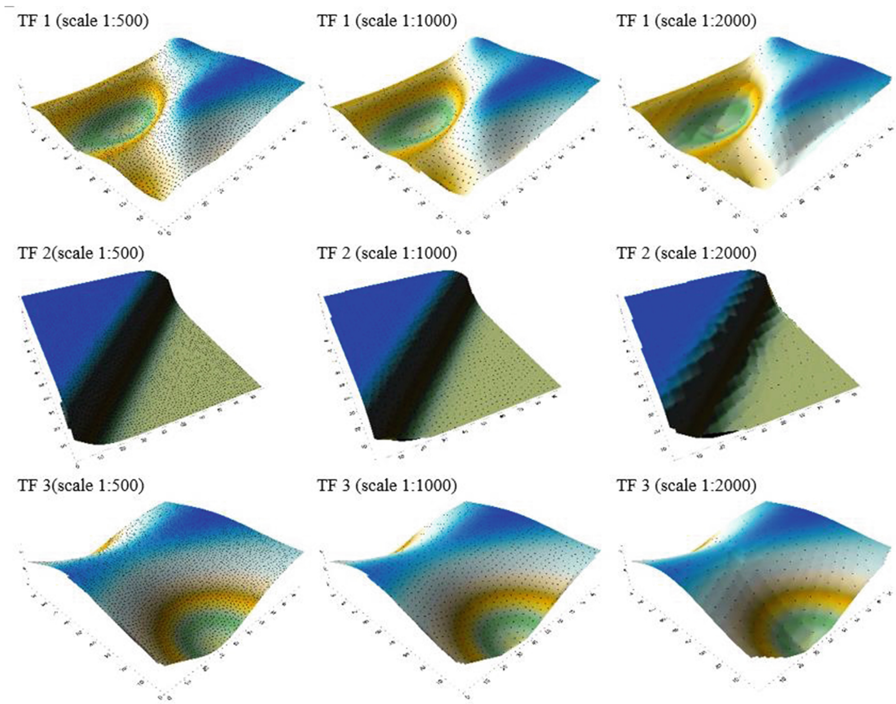


Fig. 8. The surfaces with spatial distribution of geodata points for each scale

the shape of tested area. For other scales on surfaces, you can determine their roughness, which is associated with the number of points in the set. However, they do not differ much from the reference surface.

During the usage of created method for the 1:500 scale, 97.9% of points were reduced. For scale 1:1000, the number of points decreased by 99.3% and for scale 1:2000 the reduction covered as much as 99.8%. Analyzing and processing of geodata sets is easier and more effective. A summary of statistics for received sets of samples is given in Table 2, which contains minimum depth, maximum depth and mean depth.

For all reduced sets, the smallest depth of test dataset has been preserved. As the scale decreases, the maximum and mean values of depth decrease slightly. It is associated with the number of points in the output sets. The created method is mostly concentrated on the smallest values of depth, that are associated with the navigation safety. The usage of created reduction method makes it possible to preserve their actual 100% characteristics.

Table 2. Comparison of statistics related to the depth value for test scenarios.

	MIN depth	MAX depth	MEAN depth
TF1 - test	5.00 m	29.79 m	15.19 m
TF1 - scale 1:500	5.00 m	29.73 m	14.92 m
TF1 - scale 1:1000	5.00 m	29.55 m	14.69 m
TF1 - scale 1:2000	5.00 m	29.56 m	14.27 m
TF2 - test	4.52 m	15.11 m	9.81 m
TF2 - scale 1:500	4.52 m	15.11 m	9.71 m
TF2 - scale 1:1000	4.52 m	15.11 m	9.60 m
TF2 - scale 1:2000	4.52 m	15.11 m	9.37 m
TF3 - test	4.88 m	30.35 m	12.39 m
TF3 - scale 1:500	4.88 m	30.27 m	12.13 m
TF3 - scale 1:1000	4.88 m	30.15 m	11.88 m
TF3 - scale 1:2000	4.88 m	29.44 m	11.55 m

4 Conclusions

In this paper we presented application of neural networks for sea bottom surface modelling. Nowadays this problem is important due to application of modern sensor in remote sensing, which are able to collect big datasets. Neural networks, besides their possibility to create surface model, can also reduce data required for surface reconstruction. This kind of reduction is based on replacement of sample data by neural network structure, which can require less data to be remembered to reconstruct surface. Another approach may be to use the networks for a clustering, which can be part of the reduction process. Research does not consider informatics methods of data storing and algorithm implementation, what will decide on physical data size in files.

Based on study can be stated that the RBF neural network are more flexible in surface reconstruction, because they can be used as an interpolators or approximators. But in the first case, due to exact solution, is not possible data reduction. SLP can only be used as approximator, but is more efficient in data reduction. Reduction of neurons depends also on tolerance threshold, which value should decide on hidden neuron number. The Kohonen network used in the second stage of the original reduction method also fulfills a well-established role. Created method of reduction makes it possible to preserve the real value of depth in the real location of the measuring point. The geodata set is much reduced in a strict connection to a scale of bathymetric map. Obtained data are characterizes by the lower surface projection failure what is a result of preserving the real value of the depth without its interpolation. Bathymetric geodata obtained in the process of reduction may be implemented in the process of creation the bathymetric maps.

Studies were performed on test functions. We used low-density data and those with very high density. The first one was used to reconstruction the bottom surface, while the second one was used during the reduction. Some important notice is that neural networks need additional method, which allows implementing them for large surface modelling. This can cause less data reduction ability. This ability can varied due to implemented methods and should be considered individually.

References

1. Balicki, J., Kitowski, Z., Stateczny, A.: Artificial neural networks for modelling of spatial shape of sea bottom. In: IV Conference of Neural Networks and Their Applications, Zakopane (1999)
2. Bishop, C.M.: Neural Networks for Pattern Recognition. Oxford University Press, New York (1995)
3. Carlson, R.E., Foley, T.A.: Radial Basis Interpolation Methods on Track Data, Lawrence Livermore National Laboratory, No. UCRL-JC-1074238 (1991)
4. Franke, R., Nielson, G.: Smooth interpolation of large sets of scattered data. Int. J. Numer. Methods Eng. **15**, 1691–1704 (1980)
5. Franke, R.: Scattered data interpolation: test of some methods. Math. Comput. **38**(157), 181–200 (1982)
6. Haykin, S.: Neural Networks a Comprehensive Foundation. Macmillan College Publishing Company, New York (1994)
7. Kansa, E.J., Carlson, R.E.: Improved accuracy of multiquadric interpolation using variable shape parameter. Comput. Math. Appl. **24**(12), 99–120 (1992)
8. Lubczonek, J., Stateczny, A.: Concept of neural model of the sea bottom surface. In: Rutkowski, L., Kacprzyk, J. (eds.) Neural Networks and Soft Computing Book Series: Advances in Soft Computing, pp. 861–866. Physica, Heidelberg (2003). https://doi.org/10.1007/978-3-7908-1902-1_135
9. Lubczonek, J.: Hybrid neural model of the sea bottom surface. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) ICAISC 2004. LNCS (LNAI), vol. 3070, pp. 1154–1160. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24844-6_181
10. Maleika, W., Palczynski, M., Frejlichowski, D.: Interpolation methods and the accuracy of bathymetric seabed models based on Multibeam Echosounder data. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ACIIDS 2012. LNCS (LNAI), vol. 7198, pp. 466–475. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-28493-9_49
11. Smith, M., Goodchild, M., Longley, P.: Geospatial Analysis - A Comprehensive Guide to Principles, Techniques and Software Tools, 3rd edn. (2009)
12. Stateczny, A., Kazimierski, W.: Determining manoeuvre detection threshold of GRNN filter in the process of tracking in marine navigational radars. In: Kawalec, A., Kaniewski, P. (eds.) 2008 Proceedings International Radar Symposium, Wrocław, pp. 242–245 (2008)
13. Stateczny, A.: Artificial neural networks for comparative navigation. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) ICAISC 2004. LNCS (LNAI), vol. 3070, pp. 1187–1192. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24844-6_186
14. Stateczny, A.: The neural method of sea bottom shape modelling for the spatial maritime information system. In: Brebbia, CA., Olivella, J. (eds.) Maritime Engineering and Ports II. Water Studies Series, Barcelona, vol. 9, pp. 251–259 (2000)
15. Tarwater, A.E.: A parameter study of Hardy's multiquadric method for scattered data interpolation. Lawrence Livermore National Laboratory. Technical Report: UCRL-53670 (1985)

16. Wlodarczyk-Sielicka, M.: Importance of neighborhood parameters during clustering of bathymetric data using neural network. In: Dregvaite, G., Damasevicius, R. (eds.) ICIST 2016. CCIS, vol. 639, pp. 441–452. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46254-7_35
17. Wlodarczyk-Sielicka, M., Stateczny, A.: Fragmentation of hydrographic big data into subsets during reduction process. In: Proceedings of 2017 Baltic Geodetic Congress (Geomatics), 22–25 June 2017, Poland. Gdansk University of Technology (2017). https://doi.org/10.1109/bgc.geomatics.2017.67
18. Wlodarczyk-Sielicka, M., Stateczny, A.: General concept of reduction process for big data obtained by interferometric methods. In: Rohling, H. (ed.) Proceedings of 18th International Radar Symposium (IRS 2017), Prague (2017). https://doi.org/10.23919/irs.2017.8008212. ISBN: 978-3-7369-9343-3
19. Wlodarczyk-Sielicka, M., Stateczny, A.: Clustering bathymetric data for electronic navigational charts. J. Navig. **69**(05), 1143–1153 (2016). https://doi.org/10.1017/s0373463316000035. The Royal Institute of Navigation 2016

Application of an Ant Colony Optimization Algorithm in Modeling the Heat Transfer in Porous Aluminum

Rafał Brociek[1(✉)], Damian Słota[1], Mariusz Król[2], Grzegorz Matula[2], and Waldemar Kwaśny[2]

[1] Institute of Mathematics, Silesian University of Technology, Kaszubska 23, 44-100 Gliwice, Poland
rafal.brociek@polsl.pl
[2] Institute of Engineering Materials and Biomaterials, Silesian University of Technology, Konarskiego 18A, 44-100 Gliwice, Poland

Abstract. In this paper procedure for solving inverse heat conduction problem with fractional derivative is presented. Authors present time fractional heat conduction model with Caputo derivative and Neumann, Robin boundary conditions, which can be applied to describe process of heat conduction in porous media. Based on temperature measurements, functional describing error of approximate solution is created. Considered inverse problem is transform to find minimum of created functional. In order to solve inverse problem (find unknown parameters of model) authors applied an Ant Colony Optimization (ACO) algorithm. Finally, experiment with data from porous aluminum was carried out to check effectiveness of proposed algorithm. Goal of this paper is reconstruction unknown parameters in heat conduction model with fractional derivative and show that ACO is effective algorithm and works well in these type of problems.

Keywords: Fractional heat conduction equation · Identification
Ant colony algorithm · Inverse problem

1 Introduction

In this paper we present an algorithm to solve heat conduction inverse problem based on Ant Colony Optimization algorithm [1]. Proposed model is time fractional heat conduction equation with Caputo derivative and Neumann, Robin boundary conditions. Fractional calculus can be applied to many problems in science [2–11] and to describe phenomena of subdiffusion and superdiffusion. In paper [6] authors studies the design of PID controllers for linear dynamical systems based on fractional derivative. They also used particle swarm optimization algorithm to find optimal parameters of the controllers. Fractional derivatives are widely used in describing phenomena in nature, for example paper [8] presents application of fractional calculus in biomedicine and biology. Authors show that process of diffusion of substance in human body e.g. drug

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 370–378, 2018.
https://doi.org/10.1007/978-3-319-99972-2_30

diffusion can be described within fractional derivative. In paper [11] authors present Quasi Gamma Curve (QGM) model based on fractional derivative and applied it to image enhancement. Next, authors compared proposed algorithm to the Gamma correction and Retinex algorithm.

In this paper we focus on fractional heat conduction equation with Caputo derivative. In order to solve inverse problem (find unknown parameters), firstly we need to solve direct problem. To do this implicit finite difference scheme was applied with approximation of Caputo derivative [12]. More about numerical solutions of fractional partial differential equation and fractional calculus can be found in [13, 14].

Considered inverse problem consist in identification order of derivative and heat transfer coefficient (boundary condition of third kind) in proposed model based on additional information, which in this case is temperature measurements from porous aluminum. To find sought parameters, functional describing error of approximate solution was created and minimized. The inverse problem was transformed into optimization problem. In papers [15–19], we can find an examples of heat conduction inverse problems. For example, in paper [16], authors used homotopy perturbation method to find solution of heat conduction inverse problem with Neumann boundary condition. The problem consist in reconstruction of the function describing heat flux on the boundary. Paper [15] present a Levenberg-Marquardt algorithm to solve inverse heat conduction problem. Authors show new approach to determine dumping factor, which is a significant parameter in Levenberg-Marquardt algorithm. They applied a new approach to identifying temperature-dependent thermal conductivities. In case of inverse heat conduction problems with fractional derivative, Mrurio [19] presents a simple algorithm based on space marching mollification techniques. Author recovered the boundary temperature and heat flux functions from data temperature at some interior point. In presented model Riemann-Liouville derivative is used.

To find minimum of created functional in inverse problem authors used the Ant Colony Optimization (ACO) algorithm [1], which belongs to swarm intelligence group of algorithms. ACO algorithm is inspired by behavior of swarm of ants, which are regarded as intelligent community. Generally, swarm intelligence algorithms have wide range of application [1–7, 20–28].

This paper is organized as follows. In Sect. 2 authors present model – time fractional heat conduction equation with Neumann and Robin boundary conditions and Caputo fractional derivative. Inverse problem consist on determining heat transfer coefficient and order of fractional derivative based on temperature measurements from porous aluminum. The problem under consideration is reduced to the optimization problem. Short description of finding solution of direct problem is presented in Sect. 3. More about solution of direct problem is presented in [12]. Section 4 presents numerical example with conclusions.

2 Problem Statement

In this section we present a model which describe process of heat conduction in porous media. Let us consider the following time fractional heat conduction equation:

$$c\varrho \frac{\partial^{\alpha} u(x,t)}{\partial t^{\alpha}} = \lambda \frac{\partial^{2} u(x,t)}{\partial x^{2}}, \tag{1}$$

where $x \in (0, L_x)$, $t \in (0, T)$, α is an order of fractional derivative and c, ϱ, λ denotes specific heat, density and thermal conductivity coefficient. State function u describes distribution of the temperature. To the Eq. (1) an initial-boundary conditions are added:

$$\begin{aligned} u(x,0) &= f(x), \qquad x \in [0, L_x], \\ -\lambda \tfrac{\partial u(0,t)}{\partial x} &= q(t), \qquad t \in (0, T), \\ -\lambda \tfrac{\partial u(L_x,t)}{\partial x} &= h(t)(u(L_x, T) - u^{\infty}), \qquad t \in (0, t^{*}), \end{aligned}$$

where function q is a heat flux, function h is heat transfer coefficient, function f defines the initial condition and u^{∞} is ambient temperature. Fractional derivative occurring in left side of Eq. (1) is defined in Caputo sense as follows (for $\alpha \in (0,1)$):

$$\frac{\partial^{\alpha} u(x,t)}{\partial t^{\alpha}} = \frac{1}{\Gamma(1-\alpha)} \int_0^t \frac{\partial u(x,s)}{\partial s} (t-s)^{-\alpha} \; ds, \tag{2}$$

where Γ is the Gamma function.

Function h and order of derivative α in presented model are unknown parameters, which will be identifying. Heat transfer coefficient h is searched in form:

$$h(t) = a_1 t^2 + a_2 t + a_3, \tag{3}$$

and $\alpha = a_4$ is a constant from interval $(0.01, 0.99)$. Considered inverse problem is consist on restore the parameters a_i $(i = 1,2,3,4)$ (and therefore thermal conductivity coefficient and order of fractional derivative). Additional information for inverse problem are temperature measurements from boundary of porous aluminum sample (input data) and denoted:

$$u\big(x_p, t_k\big) = \widehat{U}_k, \qquad k = 1, 2, \ldots, N,$$

where x_p is a point from boundary of the sample and N denotes number of measurements. Solving the direct problem for fixed values of the parameters a_i, we obtain an approximate values of function u in point x_p. These values will be denoted by $U_k(h, \alpha)$. Direct problem is solved using finite difference method, approximation of boundary conditions and Caputo fractional derivative. More information about numerical scheme and solution of direct problem can be found in [12]. Therefore, based on computed temperature and input data, we create functional defining the error of approximate solution:

$$F(h,\alpha) = \sqrt{\sum_{k=1}^{N}\left(U_k(h,\alpha) - \widehat{U}_k\right)^2}. \quad (4)$$

Sought parameters is searched by minimizing function (4) using Ant Colony Optimization algorithm.

3 Ant Colony Optimization Algorithm

In this section, we discuss the Ant Colony Optimization algorithm, which is used to find minimum of functional (4). The swarm of ants as a smart society communicates with each other by leaving a pheromone spots in order to find the source of food in most effective way. In presented algorithm points from $\mathbb{R}^n$ space (where n is number of sought parameters a_i) are played role of pheromone spots. Ants leave stronger pheromone spots in points where functional (4) has lower values. More information on the described algorithm can be found in the papers [1, 3]. In addition, the presented algorithm has been adapted to calculations on many threads.

To describe ACO algorithm we assume following symbols:

F	– minimized function (objective function),
nT	– number of threads,
$M = nT \cdot k$	– number of ants ($k \in \mathbb{Z}$),
L	– number of pheromone spots,
I	– number of iterations,
$\xi = 1.0, q = 0.9$	– parameters of algorithm.

ACO algorithm

Initialization of the algorithm

1. Setting parameters of the algorithm L, M, I, nT.
2. Generating L pheromone spots (solutions) in random way and creating the initial archive solutions T_0.
3. Computing values of minimized function for every pheromone spot (solution) and assigning rank to the solutions referring to the value of the minimized function. Sort elements in T_0, according to their qualities (descending).

Iterative process

4. Assigning the probabilities to the pheromone spots (solutions) according to the formula:

$$p_l = \frac{\omega_l}{\sum_{l=1}^{L}\omega_l} \qquad l = 1, 2, \ldots, L,$$

where ω_l is the weight associated to the l-th solution (solution which rank is l) and expressed by the formula

$$\omega_l = \frac{1}{qL\sqrt{2\pi}} e^{\frac{-(l-1)^2}{2q^2L^2}}.$$

5. The ant chooses the l-th solution according to probabilities p_l.
6. The ant transforms the j-th (j = 1,2, ..., n) coordinate of the l-th solution s_j^l by sampling the neighborhood using the probability density function (Gaussian function):

$$g(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}},$$

where $\mu = s_j^l, \quad \sigma = \frac{\xi}{L-1}\sum_{p=1}^{L}\left|s_j^p - s_j^l\right|$.

7. Steps 5–6 are repeated for each ant. Hence, we obtain M new solutions (pheromone spots).
8. Dividing the population on nT groups (groups will be calculated in parallel way).
9. Determination the value of minimized function for each new solution in population (parallel calculation).
10. Assign rank to the elements from T_i and sort it.
11. Repeating steps 4–10 I times.
12. Return best solution from last archive $T_{i.}$

For fixed values of parameters L, M, I number of calculation of minimized function F is equal to $L + M \cdot I$.

4 Numerical Experiment

Input data for inverse problem $\widehat{U}_k$ (measured temperatures) is obtained from sample of porous aluminum. The sample was made by pressure the powders' of aluminum in the plate hydraulic press at 150 bar. Sample was heated to 573 K at speed of 1 K/s and then cooled to ambient temperature. During that time, sample temperature measurements were done. Heat transfer coefficient and order of derivative are unknown parameters, other data has values as follows:

$t \in [0, 71.82], x \in [0, 3.825], c = 900, \varrho = 2106, u^\infty = 298, f(x) = 573, \quad q(t) = 0, \lambda = 184.806$. Heat transfer coefficient h is sought in the form:

$$h(t) = a_1t^2 + a_2t + a_3,$$

while order of derivative $\alpha = a_4$ is constant from interval $(0, 1)$. Grid used to calculations has size 100×3990 ($\Delta x = 0.03825, \Delta t = 0.036$).

The following parameters were initialized in the ACO algorithm:

$$a_1 \in [-10, 10],\quad a_2 \in [-5, 5],\quad a_3 \in [70, 200],\quad a_4 \in [0.01, 0.99],$$
$$L = 12,\quad M = 16,\quad I = 55,\quad nT = 4$$

The ACO algorithm is probabilistic, so we decide to execute it ten times to check the stability. Parameters for the ACO algorithm were selected experimentally.

Table 1 presents values of identified parameters $a_i (i = 1, 2, 3, 4)$. Order of derivative has value 0.16. In Table 2 we can see errors of reconstructed temperature in measurement point. Average relative error is equal 1.02%, while maximal relative error is equal 3.46%.

Table 1. Results of identification parameters a_i, (σ_{a_i}- standard deviation $i = 1, 2, 3, 4$)

$\boldsymbol{a}_1$	1.87
$\boldsymbol{\sigma}_{\boldsymbol{a}_1}$	1.65
$\boldsymbol{a}_2$	4.99
$\boldsymbol{\sigma}_{\boldsymbol{a}_2}$	2.59
$\boldsymbol{a}_3$	192.71
$\boldsymbol{\sigma}_{\boldsymbol{a}_3}$	51.19
$\boldsymbol{a}_4$	0.16
$\boldsymbol{\sigma}_{\boldsymbol{a}_4}$	0.05
value of function $\boldsymbol{F}$	357.65

Table 2. Errors of temperature reconstruction (Δ_{avg}—average absolute error, Δ_{max}—maximal absolute error, δ_{avg}—average relative error, δ_{max}—maximal relative error)

$\boldsymbol{\Delta_{avg}}[\boldsymbol{K}]$	4.77
$\boldsymbol{\Delta_{max}}[\boldsymbol{K}]$	12.38
$\boldsymbol{\delta_{avg}}[\%]$	1.02
$\boldsymbol{\delta_{max}}[\%]$	3.46

Figure 1 shows how computed temperature from presented model fit to measured data and in the Fig. 2 distribution of errors is presented. Presented model fit well to the measured data. Maximal error is equal approximately 12 K in beginning and end of the process.

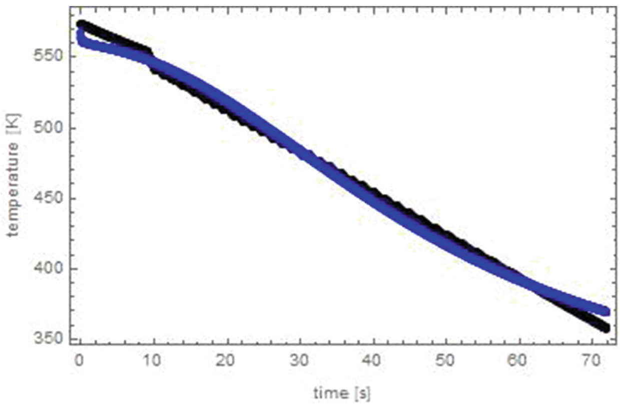


Fig. 1. Measured data (black line) and reconstructed temperature (blue line) (Color figure online)

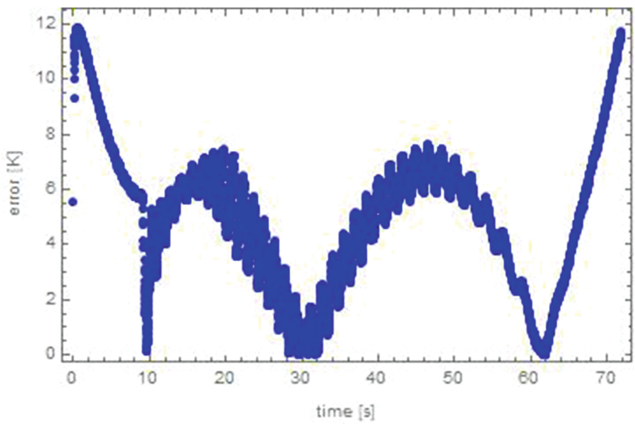


Fig. 2. Distribution of errors of reconstructed temperature

5 Conclusions

Paper presents model of heat conduction with Neumann, Robin boundary conditions and Caputo fractional derivative. This type of model can be used to modelling process of heat transfer in porous media. Authors present algorithm to solve inverse heat conduction problem based on swarm intelligent algorithm – ACO. The reconstructed temperature distribution fits sufficiently well to the measured data from porous aluminum sample The authors, in future scientific papers, plan to compare different models of heat conduction based on different fractional derivatives. In particular, models with Riemann-Liouville fractional derivative and classical derivative will be investigated.

References

1. Socha, K., Dorigo, M.: Ant colony optimization in continuous domains. Eur. J. Oper. Res. **185**, 1155–1173 (2008)
2. Brociek, R., Słota, D.: Reconstruction of the Robin boundary condition and order of derivative in time fractional heat conduction equation. Math. Model. Nat. Phenom. **13** (2018). Article no 5
3. Brociek, R., Słota, D.: Application of real ant colony optimization algorithm to solve space fractional heat conduction inverse problem. In: Dregvaite, G., Damasevicius, R. (eds.) ICIST 2016. CCIS, vol. 639, pp. 369–379. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46254-7_29
4. Brociek, R., Słota, D.: Reconstruction of the boundary condition for the heat conduction equation of fractional order. Therm. Sci. **19**, 35–42 (2015)
5. Chen, S., Liu, F., Jiang, X., Turner, I., Burrage, K.: Fast finite difference approximation for identifying parameters in a two-dimensional space-fractional nonlocal model with variable diffusivity coefficients. SIAM J. Numer. Anal. **56**, 606–624 (2016)
6. Dabiri, A., Moghaddam, P.B., Machado, J.A.T.: Optimal variable-order fractional PID controllers for dynamical systems. J. Comput. Appl. Math. **339**, 40–48 (2018)
7. Das, S.: Functional Fractional Calculus for System Identification and Controls. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-72703-3
8. Ionescu, C., Lopes, A., Copot, D., Machado, J.A.T., Bates, J.H.T.: The role of fractional calculus in modelling biological phenomena: a review. Commun. Nonlin. Sci. Numer. Simul. **51**, 141–159 (2017)
9. Ismailov, M.I., Cicek, M.: Inverse source problem for a time-fractional diffusion equation with nonlocal boundary conditions. Appl. Math. Model. **40**, 4891–4899 (2016)
10. Obrączka, A., Kowalski, J.: Modeling the distribution of heat in the ceramic materials using fractional differential equations. In: Szczygieł, M. (eds.) Materiały XV Jubileuszowego Sympozjum „Podstawowe Problemy Energoelektroniki, Elektromechaniki i Mechatroniki", PPEEm 2012, Archiwum Konferencji PTETiS, Komitet Organizacyjny Sympozjum PPEE i Seminarium BSE, vol. 32, pp. 132–133 (2012). (in polish)
11. Xiaoran, L., Shangbo, Z., Hua, L., Hongbin, T., Ying, Q.: Rhythm oscillation in fractional-order Relaxation oscillator and its application in image enhancement. J. Comput. Appl. Math. **339**, 69–84 (2018)
12. Brociek, R.: Implicit finite difference method for time fractional diffusion equations with mixed boundary conditions. Zeszyty Naukowe Politechniki Śląskiej, Matematyka Stosowana **4**, 73–87 (2014)
13. Klafter, J., Lim, S., Metzler, R.: Fractional Dynamics: Resent Advances. World Scientific, New Jersey (2012)
14. Podlubny, I.: Fractional Differential Equations. Academic Press, San Diego (1999)
15. Cui, M., Zhao, Y., Xu, B., Gao, X.: A new approach for determining damping factors in Levenberg-Marquardt algorithm for solving an inverse heat conduction problem. Int. J. Heat Mass Transf. **107**, 747–754 (2017)
16. Hetmaniok, E., Nowak, I., Słota, D., Wituła, R., Zielonka, A.: Solution of the inverse heat conduction problem with Neumann boundary condition by using the homotopy perturbation method. Therm. Sci. **17**, 643–650 (2013)

17. Hetmaniok, E., Słota, D., Zielonka, A.: Parallel procedure based on the swarm intelligence for solving the two-dimensional inverse problem of binary alloy solidification. In: Wyrzykowski, R., Deelman, E., Dongarra, J., Karczewski, K., Kitowski, J., Wiatr, K. (eds.) PPAM 2015. LNCS, vol. 9574, pp. 287–297. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-32152-3_27
18. Hetmaniok, E.: Inverse problem for the solidification of binary alloy in the casting mould solved by using the bee optimization algorithm. Heat Mass Transf. **52**, 1369–1379 (2016)
19. Murio, D.: Stable numerical solution of a fractional-diffusion inverse heat conduction problem. Comput. Math. Appl. **53**, 1492–1501 (2007)
20. Birvinskas, D., Jusas, V., Martisius, I., Damasevicius, R.: Data compression of EEG signals for artificial neural network classification. Inf. Technol. Control **42**, 238–241 (2013)
21. Das, R., Akay, B., Singla, R.K., Singh, K.: Application of artificial bee colony algorithm for inverse modelling of a solar collector. Inverse Prob. Sci. Eng. **25**, 887–908 (2017)
22. Jafrasteh, B., Fathianpour, N.: A hybrid simultaneous perturbation artificial bee colony and back-propagation algorithm for training a local linear radial basis neural network on ore grade estimation. Neurocomputing **235**, 217–227 (2017)
23. Woźniak, M., Połap, D.: Hybrid neuro-heuristic methodology for simulation and control of dynamic systems over time interval. Neural Netw. **93**, 45–56 (2017). https://doi.org/10.1016/j.neunet.2017.04.013
24. Karaboga, D., Akay, B.: A comparative study of artificial bee colony algorithm. Appl. Math. Comput. **214**, 108–132 (2009)
25. Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. J. Glob. Optim. **39**, 459–471 (2007)
26. Połap, D., Woźniak, M.: Polar bear optimization algorithm: meta-heuristic with fast population movement and dynamic birth and death mechanism. Symmetry **9**(10), 203:1–203:20 (2017). https://doi.org/10.3390/sym9100203
27. Woźniak, M., Połap, D.: Adaptive neuro-heuristic hybrid model for fruit peel defects detection. Neural Netw. **98**, 16–33 (2018). https://doi.org/10.1016/j.neunet.2017.10.009
28. Połap, D., Kęsik, K., Woźniak, M., Damaševičius, R.: Parallel technique for the metaheuristic algorithms using devoted local search and manipulating the solutions space. Appl. Sci. **8**(2), 293:1–293:25 (2018). https://doi.org/10.3390/app8020293

Application of the Taylor Transformation to the Systems of Ordinary Differential Equations

Radosław Grzymkowski and Mariusz Pleszczyński(✉)

Institute of Mathematics, Silesian University of Technology, Kaszubska 23, 44-100 Gliwice, Poland
mariusz.pleszczynski@polsl.pl

Abstract. In the paper the Taylor transformation is applied to systems of ordinary differential equations, including nonlinear differential equations. Apart from the description of the method, its computational effectiveness is demonstrated on example. Efficiency of the proposed method is confirmed it with the selected classical methods devoted to problems of considered kind. The present paper is an introduction to some further research in this area, which is very important for a wide range of problems described by means of the systems of ordinary differential equations.

Keywords: Taylor transformation · System of ordinary differential equations

1 Introduction

Due to the development of electronic computational technique, including the technique of symbolic computations, some new possibilities for solving the initial problems for ordinary differential equations have been achieved. Some importance is gained by the specific methods which earlier, without the possibility of symbolic computations, could be considered only theoretically. Group of such methods includes the approach based on one of the fundamental theorems in mathematical analysis, known in the professional literature as the Taylor formula [4, 6, 8, 15, 17]. Idea of this method is very simple, but its application requires calculation of the higher order derivatives of the objective function which lead to a problem of symbolic computations with the use of digital machines not available yet. The symbolic computational techniques for many years were used in solving the differential equations [2, 14], but the approach applied in this elaboration is a little bit different than in other procedures. The approach presented in this paper has one very strong advantage - by regulating the number of terms in the series we can obtain the approximate solution arbitrarily close, within the numerical precision, to the exact solution. In our days, when we have highly developed computer techniques at our disposal, such a problem can be resolved. There exist many methods devoted to the discussed initial problems, like, for example, the well known group of Runge-Kutta methods [3, 7, 12, 15], or the less known Adomian's decomposition method [1, 10–13]. In this paper we compare the effectiveness of the investigated method based on the Taylor formula with the effectiveness of the sixth order Runge-

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 379–387, 2018.
https://doi.org/10.1007/978-3-319-99972-2_31

Kutta method and the numerical method implemented in Mathematica software [5, 18] for the systems of ordinary differential equations. Efficiency of this method for single ordinary differential equations was demonstrated in [9]. Presented paper introduces the foundations for further research in this field, which is already in progress and the results will be published in the nearest future.

2 Theoretical Backgrounds

We consider the following system of differential equations

$$u_i'(x) = f_i\big(x, u_1(x), u_2(x)\ldots, u_p(x)\big) := f_i\big(x, \mathbf{u}_p(x)\big), i = 1, 2, \ldots, p \tag{1}$$

with initial conditions

$$u_i(x_0) = u_i^0, \quad i = 1, 2, \ldots, p, \tag{2}$$

in which the variable x belongs to the interval $\langle x_0, x_0 + \delta\rangle$, $x_0 \in \mathbb{R}$, $0 < \delta \in \mathbb{R}$, $u_i \in \mathbb{R}$, $i = 1, 2, \ldots, p$, whereas the functions u_i and f_i are of class C_1 and C_n, respectively (C_1 is the class of continuous functions with the continuous first derivative and C_n is the class of continuous functions with the continuous partial derivatives of orders 1 to n).

Under the taken assumptions we have from the Taylor formula

$$\begin{aligned} f_i\big(x, \mathbf{u}_p(x)\big) &= \sum_{k=0}^{n-1} \frac{F_{ik}\big(x_0, \mathbf{u}_p(x_0)\big)}{k!}(x - x_0)^k \\ &+ \frac{F_{in}\big(x_0 + \theta(x - x_0), \mathbf{u}_p(x_0 + \theta(x - x_0))\big)}{n!}(x - x_0)^n, \qquad i = 1, 2, \ldots, p, \end{aligned} \tag{3}$$

where $\theta \in (0, 1)$, and the functions F_{ik}, $i = 1, 2, \ldots, p$, $k = 0, 1, \ldots, n$, are determined by relations

$$F_{ik}\big(x, \mathbf{u}_p(x)\big) = \left\{ \left[\frac{d}{dx} F_{ik-1}\big(x, \mathbf{u}_p(x)\big)\right]_{u_l'(x) = f_l\left(x, \mathbf{u}_p(x)\right), l = 1, 2, \ldots, p} \right\}, \tag{4}$$

$i = 1, 2, \ldots, p, k = 0, 1, \ldots, n$. For example, if $f_i = f_i(x, u_1(x), u_2(x))$, $i = 1, 2$, then we have

$$F_{10} = f_1, F_{20} = f_2,$$

$$F_{11} = f_2 f_1^{(0,0,1)} + f_1 f_1^{(0,1,0)} + f_1^{(1,0,0)}, F_{21} = f_2 f_2^{(0,0,1)} + f_1 f_2^{(0,1,0)} + f_2^{(1,0,0)},$$

$$
\begin{aligned}
F_{12} = f_1^{(0,1,0)}\left(f_2 f_1^{(0,0,1)} + f_1 f_1^{(0,1,0)} + f_1^{(1,0,0)}\right) + f_2\left(f_2 f_1^{(0,0,2)} + f_1 f_1^{(0,0,1)} + f_1^{(1,0,1)}\right) \\
+ f_1 f_1^{(1,0,1)} + f_1^{(0,0,1)}\left(f_2 f_2^{(0,0,1)} + f_1 f_2^{(0,1,0)} + f_2^{(1,0,0)}\right) + f_2 f_1^{(1,0,1)} \\
+ f_2\left(f_2 f_1^{(0,0,2)} + f_1 f_1^{(0,0,1)} + f_1^{(1,0,1)}\right) + f_1 f_1^{(1,0,1)} \\
+ f_1\left(f_2 f_1^{(0,1,1)} + f_1 f_1^{(0,2,0)} + f_1^{(1,1,0)}\right) + f_1^{(2,0,0)},
\end{aligned}
$$

$$
\begin{aligned}
F_{22} = f_2^{(0,1,0)}\left(f_2 f_1^{(0,0,1)} + f_1 f_1^{(0,1,0)} + f_1^{(1,0,0)}\right) + f_2^{(0,0,1)}\left(f_2 f_2^{(0,0,1)} + f_1 f_2^{(0,1,0)} + f_2^{(1,0,0)}\right) \\
+ f_2 f_2^{(1,0,1)} + f_2\left(f_2 f_2^{(0,0,2)} + f_1 f_2^{(0,1,1)} + f_2^{(1,0,1)}\right) + f_1 f_2^{(1,1,0)} \\
+ f_1\left(f_2 f_2^{(0,1,1)} + f_1 f_2^{(0,2,0)} + f_2^{(1,1,0)}\right) + f_1^{(2,0,0)},
\end{aligned}
$$

and so on, where

$$
f_i^{(\alpha,\beta,\gamma)} = \frac{\partial f_i^{\alpha+\beta+\gamma}(x, u_1(x), u_2(x))}{\partial x^\alpha \partial u_1^\beta \partial u_2^\gamma}, i = 1, 2,
$$

for $\alpha, \beta, \gamma \in \{0, 1, \ldots, n\}$ and $\alpha + \beta + \gamma \leq n$.

The presented calculations do not look too much encouragingly, indeed, especially for the higher values of n. However in practice having at disposal some computer tool for symbolic calculations, like for instance the computational platform *Mathematica 8*, the derivation of the functions F_{ik}, $i = 1, 2, \ldots, p$, $k = 0, 1, \ldots, n$, becomes very simple. We may realize this with Program 1. The corresponding source code is presented below.

```
(* Program 1 *)
program1[f1_,f2_,x0_,u10_,u20_,n_]:=Module[{ty,tpf1,tpf2},
tpf1=Table[f1[x,u1[x],u2[x]],{n}]; tpf2=Table[f2[x,u1[x],u2[x]],{n}];
Do[tpf1[[i]]=D[tpf1[[i-1]],x]; tpf2[[i]]=D[tpf2[[i-1]],x];
tpf1[[i]]=tpf1[[i]]/.{u1'[x]->tpf1[[1]],u2'[x]->tpf2[[1]]};
tpf2[[i]]=tpf2[[i]]/.{u1'[x]->tpf1[[1]],u2'[x]->tpf2[[1]]}, {i,2,n}];
tpf1=tpf1/.{u1[x]->u10,u2[x]->u20,x->x0};
tpf2=tpf2/.{u1[x]->u10,u2[x]->u20,x->x0}; Return[{tpf1,tpf2}]];
```

The above program for: functions $f_i\left(x, \mathbf{u}_p(x)\right)$, $i = 1, 2, \ldots, p$, point x_0 and conditions $u_i(x_0) = u_i^0$, $i = 1, \ldots, p$, $k = 0, 1, \ldots, n$, determines the values $F_{ik}^0 = F_{ik}(x, \mathbf{u}_p(x_0))$, $i = 1, 2, \ldots, p$, $k = 0, 1, \ldots, n$. The procedure is illustrated by Example 1.

Example 1. *For functions* $f_1(x, u_1(x), u_2(x)) = -1 - 3x - u_1(x)u_2(x)$ *and* $f_2(x, u_1(x), u_2(x)) = x - 1 + u_1(x)\sin^2 x^2 - u_2(x)\cos x$, *and conditions* $u_1(0) = 1$, $u_2(0) = -1$ *for* $n = 9$, *by referring to Program 1 with instructions*

```
f1[x_,u1_,u2_]:=-1-3x-u1*u2;
f2[x_,u1_,u2_]:=x-1+u1*Sin[x]^2-u2*Cos[x];
program1[f1,f2,0,1,-1,10]
```

we obtain the following values of the functions F_{1k} and F_{2k}, $k = 0, 1, \ldots, 9$:

k	0	1	2	3	4	5	6	7	8	9
F_{1k}	0	−3	−3	−4	14	91	194	−2730	−17810	−28311
F_{2k}	0	1	0	−37	−43	299	632	2713	1817	−372061

Using (1) and (2) we obtain:

$$\begin{aligned} u_i'(x) &= \sum_{k=0}^{n-1} \frac{F_{ik}^0}{k!}(x-x_0)^k \\ &+ \frac{F_{in}\left(x_0+\theta(x-x_0), \mathbf{u}_p(x_0+\theta(x-x_0))\right)}{n!}(x-x_0)^n, i = 1, 2, \ldots, p. \end{aligned} \tag{5}$$

If we divide the interval $\langle x_0, x_0+\delta \rangle$ into m equal parts, that is we discretize this interval according to the formula

$$x_j = x_0 + jh,\ j = 0, 1, \ldots, m, \tag{6}$$

where $h = \frac{\delta}{m}$, then by integrating both sides of relation (5) in subinterval $\langle x_0, x_1 \rangle$ of interval $\langle x_0, x_0+\delta \rangle$, we get

$$\begin{aligned} \int_{x_0}^{x_1} u_i'(t)dt \quad &= \sum_{k=0}^{n-1} \frac{F_{ik}^0}{k!} \int_{x_0}^{x_1} (t-x_0)^k dt \\ &+ \int_{x_0}^{x_1} \frac{F_{in}\left(x_0+\theta(x-x_0), \mathbf{u}_p(x_0+\theta(x-x_0))\right)}{n!}(t-x_0)^n, \\ &i = 1, 2, \ldots, p. \end{aligned} \tag{7}$$

If we assume that the functions f_i, $i = 1, 2, \ldots, p$, and their partial derivatives in the considered closed region are bounded, then we have

$$\forall n \in \mathbb{N} \exists M_{in} \in \mathbb{R}^+ \forall x \in \langle x_0, x_0+\delta \rangle : |F_{in}| \le M_{in},$$

which leads to the conclusion that the relation (7) yields

$$\left| u_i^1 - u_i^0 - \sum_{k=0}^{n-1} \frac{F_{ik}^0}{(k+1)!} h^{k+1} \right| \le \frac{M_{in} h^{n+1}}{(n+1)!}, \qquad i = 1, 2, \ldots, p. \tag{8}$$

From relation (8) it follows immediately that we can determine the approximate values u_i^1 of functions u_i, $i = 1, 2, \ldots, p$ within the accuracy of $\Delta_{i1} = \frac{M_{in} h^{n+1}}{(n+1)!}$. Thus we have

$$u_i^1 = u_i^0 + \sum_{k=0}^{n-1} \frac{F_{ik}^0}{(k+1)!} h^{k+1}, \qquad i = 1, 2, \ldots, p. \tag{9}$$

Applying the same arguments in formula (9) for each subinterval $\langle x_j, x_{j+1} \rangle$, $j = 0, \ldots, m-1$, of interval $\langle x_0, x_0 + \delta \rangle$, we get

$$u_i^{j+1} = u_i^j + \sum_{k=0}^{n-1} \frac{F_{ik}^j}{(k+1)!} h^{k+1} \quad i = 1, 2, \ldots, p, \quad j = 0, 1, \ldots, m-1, \tag{10}$$

where $F_{ik}^j = F_{ik}\big(x_j, \mathbf{u}_p\big(x_j\big)\big)$, and $u_i^j, j = 1, 2, \ldots, m$, $i = 1, 2, \ldots, p$, are the approximate values of functions u_i, $i = 1, 2, \ldots, p$, at points x_j, $j = 1, \ldots, m$, and these values are determined with the error not greater than $\Delta_{ij} = \frac{jM_{in}h^{n+1}}{(n+1)!}$, $i = 1, 2, \ldots, p, j = 1, 2, \ldots, m$. Parameter n, deciding about the precision of the results, can be treated as the order of investigated method.

3 Example

In order to demonstrate the efficiency of the proposed method we present now an example of the problem (1) and (2). The solution obtained will be compared with the exact solution and also with the other approximate solutions, that is with the solution obtained by using the sixth order Runge-Kutta method and the method built in *Mathematica* platform (and executed by calling the instruction NDSolve).

We consider the system of Eqs. (1) in which $f_1 = 1 - 3x\sin x - u_1 - u_2, f_2 = u_2 - x^2\cos x + u_1$, for $x \in [0, \pi]$, with conditions (2): $u_1(0) = 0$, $u_2(0) = 0$. Analytical solution of such defined problem is as follows

$$u_1(x) = \tfrac{1}{2}(-x^2 - 2x^2\cos x + 2x + 2x\sin x - 6\sin x + 6x\cos x),$$
$$u_2(x) = \tfrac{1}{2}(x^2 - 2x^2\sin x + 2x^2\cos x - 2x\sin x + 4\sin x - 4x\cos x).$$

Dividing the interval $[0, \pi]$ into $m = 10$ equal parts and taking to the sum given in (10) seven elements ($n = 6$), we obtain the results presented in Figs. 1, 2 and 3. The first figure displays the following solutions: the exact one (solid line), obtained by using the sixth order Runge-Kutta method (squares), obtained by applying the *Mathematica* platform (diamonds) and finally obtained by using the examined method (stars), for function $u_1(x)$ – left figure and for function $u_2(x)$ – right figure. Figure 1 shows the comparison of the absolute errors Δ of results obtained by applying all the methods (analogous notation and arrangement of plots is used as before).

Since the errors obtained by using the Runge-Kutta method are significantly greater than the errors generated by other methods, we present in Fig. 1 the comparison of only three methods except at the Runge-Kutta method.

It is obvious that the method should give better results for the increasing value of parameter m as well as for the increasing value of parameter n. We illustrate this trend

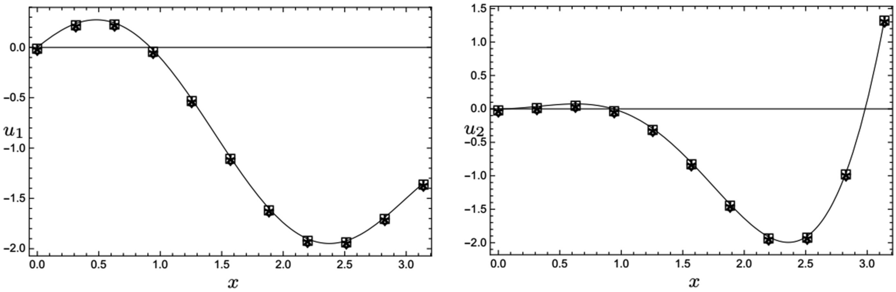


Fig. 1. Solutions (exact – solid line, sixth order Runge-Kutta method – □, procedure from *Mathematica* – ◇, examined method – $*$) for function $u_1(x)$ (left figure) and $u_2(x)$ (right figure)

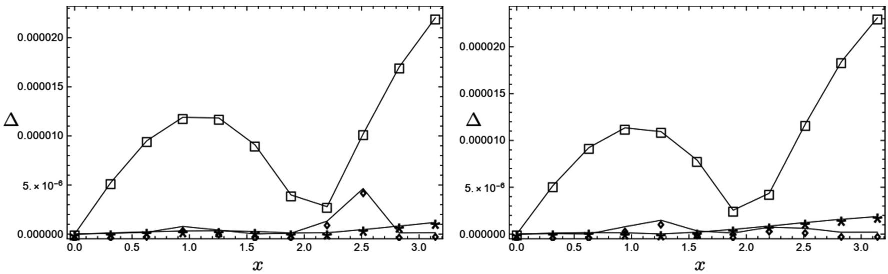


Fig. 2. Absolute errors for the case of $m = 10$ and $n = 6$ for function $u_1(x)$ (left figure) and $u_2(x)$ (right figure)

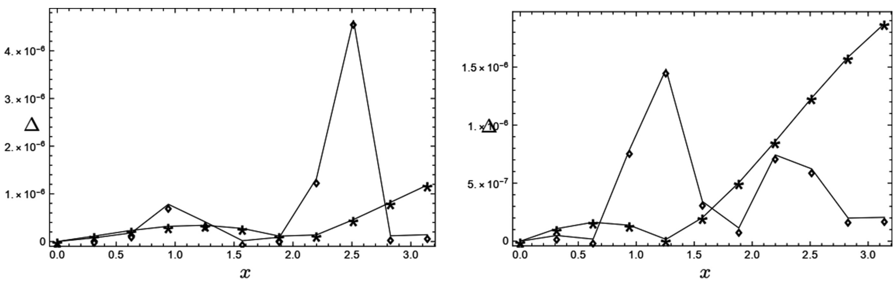


Fig. 3. Absolute errors for the case of $m = 10$ and $n = 6$ for function $u_1(x)$ (left figure) and $u_2(x)$ (right figure)

in the next two figures – in the first one we increase m without changing n and in the second one, inversely, we increase n without changing m (Figs. 4 and 5).

As we can see, in both cases, that is by increasing the dense of discretization as well as by increasing the order of discussed method, we obtain much better reconstruction of the exact solution than by using the classical Runge-Kutta method or the method implemented in *Mathematica* platform.

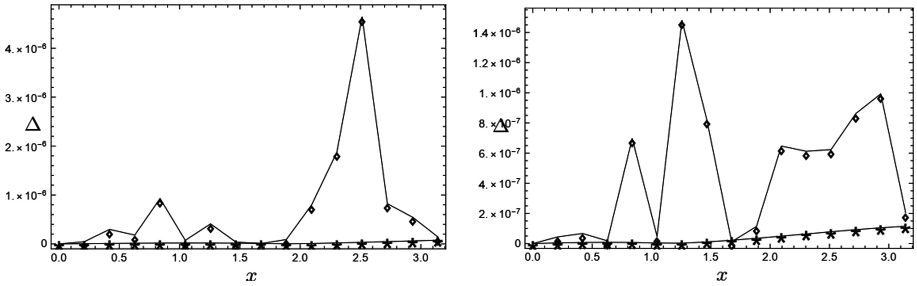


Fig. 4. Absolute errors for the case of $m = 15$ and $n = 6$ for function $u_1(x)$ (left figure) and $u_2(x)$ (right figure)

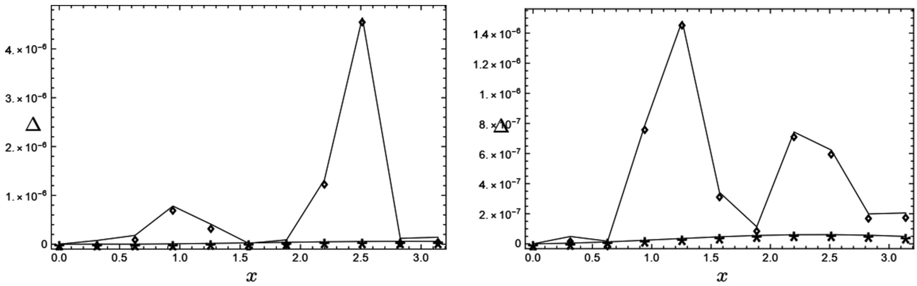


Fig. 5. Absolute errors for the case of $m = 10$ and $n = 7$ for function $u_1(x)$ (left figure) and $u_2(x)$ (right figure)

The other executed tests confirmed that the Runge-Kutta method is the least effective method from all the methods taken here into account, therefore we will compare in the future investigations only the examined method and the *Mathematica* build-in method.

4 Conclusions

The example, presented in this work, shows that the discussed method is effective and the solution obtained by using the proposed approach based on the Taylor transformation is significantly better, not only than the solutions obtained by applying the classical Runge-Kutta method (of the, most commonly used, fourth order but even of the sixth order), but also than the solutions received by applying the numerical method from the *Mathematica* software. Additionally, the examined method is universal. We are able to solve the systems of any reasonable number of differential equations (linear or nonlinear as well).

This paper finds its place in the branch of theoretical informatics because it allows to create the effective models operating on the basis of symbolic computations. In relation to this, the programming of applications, which models and simulates the technical and economical problems, in various programming languages can become much more simple. For preparing this paper the Mathematica language was used, however in future, in order to improve the effectiveness and usefulness in consequence, of the described methods we plan the hybridisation - that is the connection of possibilities given by Mathematica language (symbolic computations) and other classical, high efficient languages (for example, C++, C#) as well as the parallelization (which is possible for all programming languages mentioned here).

The presented paper is just the introduction to further research concerning systems of ordinary differential equations and validity of such investigations is confirmed by the wide range of applications of differential equations. In our next research we intend to discuss some systems of differential equations more advanced with respect to the number of equations as well as to their complexity. The next step of the undertaken research will be application of the proposed method to systems of algebraic-differential-integral equations and to the differential equations with delayed arguments.

References

1. Adomian, D.: A review of the decomposition method in applied mathematics. J. Math. Anal. Appl. **135**, 501–544 (1988)
2. Bikulčienė, L., Marcinkevičius, R., Navickas, Z.: Computer realization of the operator method for solving of differential equations. In: Li, Z., Vulkov, L., Waśniewski, J. (eds.) NAA 2004. LNCS, vol. 3401, pp. 179–186. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-31852-1_20
3. Cash, J.R., Karp, A.H.: A variable order Runge-Kutta method for initial value problems with rapidly varying right-hand sides. ACM Trans. Math. Softw. **16**(3), 201–222 (1990)
4. Chiou, J.S., Tzeng, J.R.: Application of the Taylor transform to nonlinear vibration problems. J. Vib. Acoust. **118**(1), 83–87 (1996)
5. Gliński, H., Grzymkowski, R., Kapusta, A., Słota, D.: Mathematica 8. Jacek Skalmierski Computer Studio, Gliwice (2012). (in Polish)
6. Farzad, E., Majid, G., Erfan, S., Seied, A.H.H., Gholam, R.S.: Application of the differential transformation method for nonlocal vibration analysis of functionally graded nanobeams. J. Mech. Sci. Technol. **29**(3), 1207–1215 (2015)
7. Gottlieb, S., Shu, C.-W.: Total variation diminishing Runge-Kutta schemes. Math. Comput. **67**, 73–85 (1998)
8. Grzymkowski, R.: Taylor transformation and its applications. Jacek Skalmierski Computer Studio, Gliwice (2015). (in Polish)
9. Grzymkowski, R., Hetmaniok, E., Pleszczyński, M.: A novel algorithm for solving the ordinary differential equations. In: Hetmaniok, E., Słota, D., Trawiński, T., Wituła, R. (eds.) Selected Problems on Experimental Mathematics, pp. 103–112. Silesian University of Technology Press, Gliwice (2017)
10. Grzymkowski, R., Hetmaniok, E., Słota, D.: Selected Computational Methods in the Variational Calculus and in the Differential and Integral Equations. Jacek Skalmierski Computer Studio, Gliwice (2002). (in Polish)
11. Grzymkowski, R., Pleszczyński, M., Słota, D.: Application of the Adomian decomposition method for solving the heat equation in the cast-mould heterogeneous domain. Arch. Foundry Eng. **9**(4), 57–62 (2009)

12. Grzymkowski, R., Pleszczyński, M., Słota, D.: Comparing the Adomian decomposition method and the Runge-Kutta method for solutions of the Stefan problem. Int. J. Comput. Math. **83**(4), 409–417 (2006)
13. Grzymkowski, R., Pleszczyński, M., Słota, D.: The two-phase Stefan problem solved by the Adomian decomposition method. In: Hamza, M.H. (ed.) Proceedings of the 15th IASTED International Conference on Applied Simulation and Modelling, Rhodes, Greece, pp. 511–516. Acta Press, Anaheim (2006)
14. Navickas, Z., Bikulciene, L., Rahula, M., Ragulskis, M.: Algebraic operator method for the construction of solitary solutions to nonlinear differential equations. Commun. Nonlinear Sci. Numer. Simul. **18**(6), 1374–1389 (2013)
15. Ni, Q., Zhang, Z.L., Wang, L.: Application of the differential transformation method to vibration analysis of pipes conveying fluid. Appl. Math. Comput. **217**(16), 7028–7038 (2011)
16. Tselios, K., Simos, T.E.: Runge-Kutta methods with minimal dispersion and dissipation for problems arising from computational acoustics. J. Comput. Appl. Math. **175**(1), 173–181 (2005)
17. Yu, L.-T., Chen, C.-K.: Application of Taylor transformation to the thermal stresses in isotropic annular fins. J. Therm. Stresses **21**(8), 781–809 (1998)
18. Wolfram, S.: The Mathematica Book, 5th edn. Wolfram Media, Inc., Champaign (2003)

An Introduction to UMLPDSV for Real-Time Dynamic Signature Verification

Fatima Samea(✉), Muhammad Waseem Anwar, Farooque Azam, Mehreen Khan, and Muhammad Fahad Shinwari

Department of Computer and Software Engineering, College of Electrical and Mechanical Engineering, National University of Sciences and Technology (NUST), Islamabad, Pakistan
{fatima.samea16,mehreen.khan16}@ce.ceme.edu.pk,
{waseemanwar,farooq}@ceme.nust.edu.pk,
m.fahad@nust.edu.pk

Abstract. Signatures are one of the most important behavioral biometric feature which are used to recognize an individual identity. These handwritten signatures are captured as actual input signals that are written on some electronic gadgets by the user. The divergent writing patterns of individuals primarily due to variation in style, shape and steadiness create real time challenges in differentiating real signatures from the fake ones. In order to overcome the said challenge of signature recognition, this article introduces model driven approach for dynamic signature verification. Particularly, a UMLPDSV (Unified Modeling Language Profile for Dynamic Signature Verification) has been proposed to specify the signature verification requirements at high abstraction level. This provides the basis to automatically generate target models of different machine learning tools (e.g. RapidMiner process, Matlab code etc.) to perform dynamic signature verification. The applicability of UMLPDSV has been validated through internet banking case study.

Keywords: UMLPDSV · MDA · Model-driven
Dynamic signature verification

1 Introduction

Nowadays, there is a growing demand for the security of information as well as identity verification. Such verification cannot be achieved merely by use of password protection only. Biometrics are the most popular and reliable alternative to password based security systems [1].The online verification of signatures is generally one amongst the most appropriate technologies for undertaking any biometrics because manual signatures have been used as a common mean of personal authentication as every individual has special characteristics which distinguishes him/her from others. These characteristics play a crucial role in recognizing and authorizing an individual. A substantial amount of research had already been conducted in biometrics area by using those specific characteristics of each individual for authorization as they provide the most

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 388–398, 2018.
https://doi.org/10.1007/978-3-319-99972-2_32

efficient and reliable verification system to the users. The research in the biometric field includes face and fingerprints recognition, voiceprints, signatures validation etc.

Signatures are behavioral biometric characteristics for personal identification. Various electronic devices such as smartphones, PDAs (Personal Digital Assistants) and pen tablets are used to easily capture the signatures [2]. Thus due to this flexibility; the popularity of this biometric trait has increased rapidly in previous years, particularly in the domain of banking. Different methodologies using digitizer tablets have been adopted in literature for resolving the challenge of dynamic signature verification due to numerous problems such as variations between same user signatures, various shapes of signatures etc. The aim of this research is to introduce model driven approach to address the problem of real time dynamic signature verification.

This article presents a novel UML profile to simplify the process of dynamic signature verification. Particularly, the proposed profile comprises Data Acquisition, Pre-Processing, Feature Selection and Classifier Stereotypes to model the feature selection (Writing Speed, Speed Time, Pressure, Pen Status, Stroke, Velocity, Signing Time, Total dots etc.) attributes of system under consideration. This leads to transform the developed model into target models of several machine learning tools (e.g. Rapidminer process, Matlab code etc.) to perform the real time dynamic signature verification. The application of the proposed profile is validated through internet banking case study.

The remaining paper is organized as follows. Section 2 provides background and literature review. Section 3 introduces the proposed UMLPDSV profile for dynamic signature verification. Section 4 deals with the validation of the proposed UMLPDSV applied on internet banking application. The discussion is made in Sect. 5 and Sect. 6 provides the conclusions and future work.

2 Background and Related Work

In general, signature verification is classified into two types: static (offline signature verification) and dynamic (online signature verification). Static or offline signature authentication normally uses static graphic information to verify the user. On the other hand, online signatures are captured in real time through electronic equipment. It make use of more dynamic information, such as speed, velocity, acceleration and pressure of writing a signature, which is more difficult to imitate. Hence dynamic signature verification generally has higher accuracy rate than static verification [3].

Dynamic Signature Verification is categorized into two stages i.e. Enrollment and Verification. Enrollment phase enrolls a number of reference signatures to measure variations in signature [4]. Verification phase verifies the input test signature with the set of enrolled signatures. For signature verification, the first and foremost thing is to extract features from signature data, which consist of global features and local ones. Global features describe characteristics of the entire signature, such as total signature time, strokes count, and size of signature, writing speed, pen position etc. Local features show the signature trait at a time point or during a short period, such as the local velocity and stroke angle.

The block diagram for online signature verification [5] is shown in Fig. 1, where the process starts with enrollment phase in which multiple genuine signatures are collected and pre-processing is carried out on these signature samples, then features are extracted and stored in knowledge base. The verification phase consists of data acquisition, pre-processing, feature extraction and classification. Classification is done by matching input signature features with the genuine signature features available in the knowledge base. Finally, the decision is made regarding the originality of the given signature.

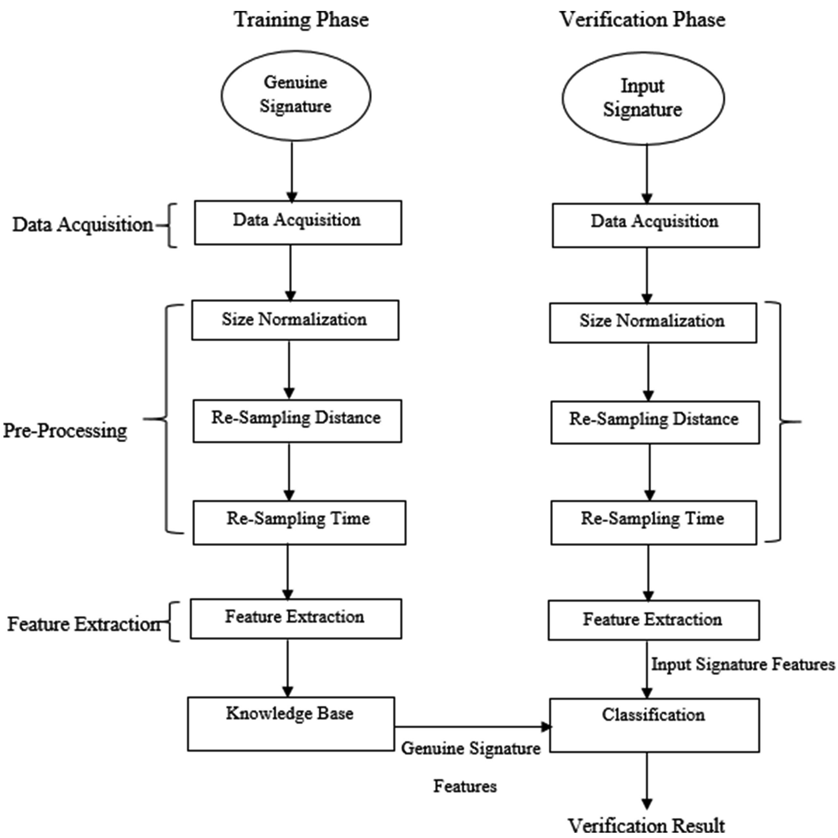


Fig. 1. Dynamic signature verification block diagram

The major requirements in dynamic signature verification are summarized below:

Data Acquisition captures the signature dynamically through a digitized device (Tablet, PDA etc.) using a stylus pen in real time. The overall signature shape is read as the sequence of points in the x-y coordinate stream. The pen enables us to gather various aspects of the signature in the form of coordinate stream in real time.

The purpose of the **pre-processing** step is to improve the performance of the authentication system. This can be achieved by size normalization, Resampling distance and Resampling time. An individual signature always varies in dimension whenever he/she signs on some paper, document or any material. Thus, there is a need to normalize the signature size in order to make an individual signature the same in size before its features are extracted. Size normalization is required to make the signatures uniformly smoothed. This will be achieved by scaling each character in horizontal and vertical direction.

In **feature extraction** stage, corresponding features of the signature are extracted which shows the length, orientation, stroke, width and height, pen up and down information etc. These features which are the global features of the signatures can help to validate the signature and are directed as follows.

(1) **Average pressure** monitors the pressure level which pen is leaving on signature area. (2) **Strongest pressure** is a local distinctive feature of the signature that can be global & unique in entire signature. (3) **Average writing speed** determines the average speed for signature. (4) **Maximum writing speed** determines the maximum speed at which user sign in. (5) **Minimum horizontal writing speed** determines the minimum writing speed from a person's signature. (6) **Time of maximum speed** represents the time required to get an individual to sign at a maximum speed. (7) **Time of minimum horizontal writing speed** is the time required by the user to sign at a minimum speed. (8) **Total signing duration** characterizes entire time required for getting user's sign, at start of signature and thus expressed in milliseconds. (9) **Signature length** is the length related to all lines which are drawn in signature and can be measured in pixels. (10) **Strokes count** is the existing lines in signature that would be measured through this feature. (11) **Pens-down count** shows the number of times the user put down the pen while signing. (12) **Pen Tilt count** counts the average pen's tilt, once the user signs on signing area. (13) **Time up ratio** determines the total signing duration when the pen was separated from the signing region. (14) **Time down ratio** defines the ratio with respect to the total signature time, once the pen had been touching the base at signature surface. (15) **Total dots recorded** expresses the total dots in signing. (16) **Average dot execution time** describes how much execution time is required for an average dot. (17) **Velocity (along x-axis)** feature shows how rapidly a person signs if only x coordinate is considered. The speed is generally expressed in total pixels in every millisecond. (18) **Velocity (along y-axis)** illustrates how rapidly a person signs if only y coordinate is considered. The speed is expressed in number of pixels per millisecond. (19) **Signature aspect ratio** describes ratio of width and height of the signature dimension expressed in pixels. These nineteen different features [6–9] are usually extracted from the signature and further classified for similarity computation.

Once all the features are extracted the final step is **classification** i.e. to make the decision on whether the signatures are real or not. For decision boundary, the classifier is trained on the signature images. Various classifiers such as Support Vector Machine (SVM), Decision trees, K means etc. are used for classification.

2.1 Related Work

Dynamic signatures are acquired as real time signals and are expressed as series of time. The variabilities in size, position and angle rotation of the same user signatures at multiple inputs could deteriorate the performance of the signature verification.

Therefore, it is very essential to align the test signatures to references before they are verified. Numerous approaches exists in literature for signature verification. Biswajit kar et al. [10] propose a new strategy for reference selection for signature authentication through SPW method. Once the feature has been selected, they evaluated it with SVM (Support Vector Machine) based verification. Chaithra and Indira [11] develop a database and recognized single stroke dynamic handwritten Kannada characters. They use K nearest neighbor classifier technique for recognizing characters. Song et al. [12] analyze the stable spectral information which is in-built in the signatures. They select the information mainly dependent on an individual for reconstructing stable spectral characteristics. Philip and Bharad [13] propose a new group of features for verification of dynamic signature. The methodology they used in their research is Webber Local Descriptor (WLD) which helped the signature verification applications to detect the forged signatures. The authors in [14] analyze the device compatibility issue for verification of dynamic signature primarily because of versatile number of different electronic devices which are being utilized nowadays. Their proposed approach was the fusion of both global and local features and the result showed the improvement for the device interoperability challenge.

In some studies, researchers have used shape context for signature verification. For example, Shin et al. [15] propose a method for verification dependent on the context of shape for inter-stoke information. They incorporate an evaluation based on average amount of pressure exerted during each stroke for signature. Results showed the effectiveness of their proposed approach. Wada and Hangai [16] introduce a method for seeking information with respect to the gripping power and position such as gripping habits which depend mainly on pen stroke direction, gripping shape of hand and so on. Julita et al. [17] discuss how to develop the dynamic signature verification process using pen and digitizer tablet and used Support Vector Machine classifier to verify that the signature is genuine or not. Fernando et al. [18] discuss a secure web based user account access and encrypted electronic documents. They design prototypes for these on windows tablet where users are enrolled by providing their signatures on tablets. They verify these on two different tablets i.e. HP and Toshiba Tablet. However, Toshiba Tablet results in better performance.

From the aforementioned literature review, it is analyzed that the dynamic signature verification is a complex process. Particularly, the selection and representation of features is difficult. Furthermore, the low level implementation is very complex. In this context, existing studies do not provide the solution to simplify the low level implementation. Therefore, there is a need to develop some solution that simplify the process of features representation and also streamline implementation complexity. To manage such limitations of existing studies, this article presents UMLPDSV. The details are given in next section.

3 UMLPDSV - UML Profile for Dynamic Signature Verification

Model Driven Architecture (MDA) is a renowned methodology that simplifies the design and verification of software applications. Therefore, it is commonly applied in different domains like embedded systems [19] etc. In MDA, a UML profile is generic extension through which models can be customized for specific domains and platforms. The profile is defined using stereotypes, tag definitions, and constraints which are applied to specific model elements, like Classes, Attributes, Operations, and Activities. Stereotypes are specific metaclasses, tagged values are standard metaattributes, and profiles are specific kinds of packages. This section represents the stereotypes and tagged value definitions for the real time dynamic signature verification profile. The proposed UMLPDSV (UML profile for Dynamic Signature Verification) is shown in Fig. 2 consisting of stereotypes and tagged values.

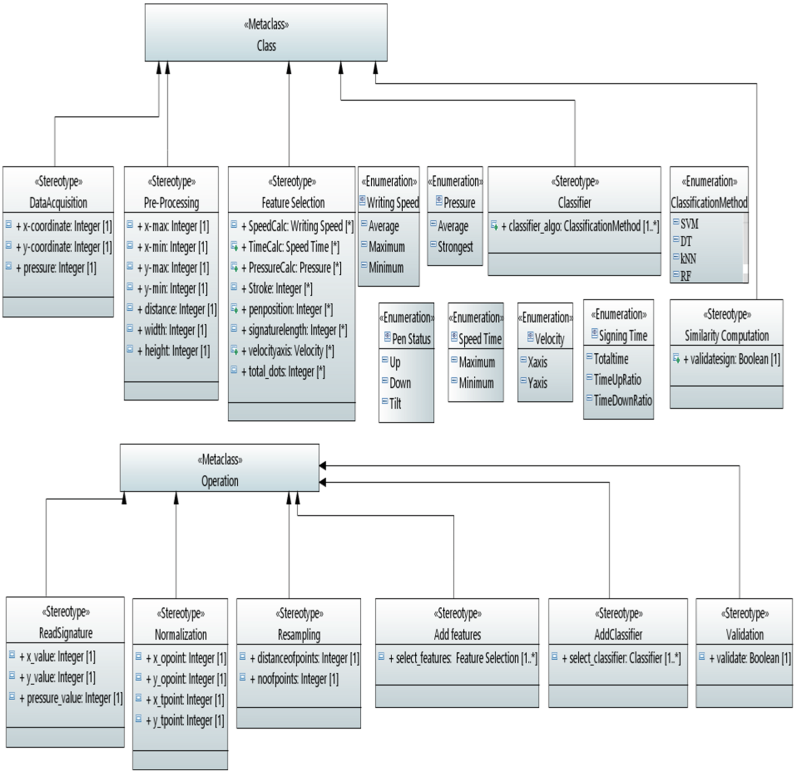


Fig. 2. Proposed UMLPDSV - UML profile for dynamic signature verification

In this metamodel, a stereotype "*DataAcquisition*" is defined in order to capture x,y coordinate and pressure values (pen is leaving on the signing area) of the signature.

This stereotype has three tag values namely x-coordinate, y-coordinate and pressure. Another stereotype is *Pre-Processing* which represents all the values required for the size normalization, re-sampling time and distance which are x-max, x-min, y-max, y-min (maximum and minimum values of x and y coordinates), distance, width and height. These values are used to normalize the signature. Since human signature have different variability patterns so normalization is important to make the signature uniform and smooth. There are two more stereotypes. The stereotype "*FeatureSelection*" defines different attributes for feature selection which are of enumeration datatype. These are: SpeedCalc, Time Calc, PressureCalc, Stroke, penposition, signature length, velocityaxis, total_dots. Another stereotype "*Classification*" selects the desired classifier for classification which is the most important step of dynamic signature verification. The various types of classifiers such as SVM (Support Vector Machine), DT (Decision trees), kNN (K nearest neighbor), RF (Random Forest) and Naïve Bayes etc. can be specified in "*Classification*" stereotype. Each of aforementioned stereotypes extends metaclass Class.

Six enumeration classes have been defined for feature selection. Enumeration class "*Writing Speed*" specifies the writing speed. The writing speed can be one of these three types: Average, Maximum, and Minimum. The second enumeration class defined in this metamodel is "*Writing Speed Time*". This class represents the writing speed time in terms of Minimum and Maximum speed time. Similarly, enumeration class "*Pen Status*" define the pen position which are Pen up, Pen down and Pen tilt. Another enumeration is "*velocity*" which defines the velocity along x and y axis (speed expressed in no of pixels per milliseconds). The enumeration "*Pressure*" specifies the pressure moment as Average and Strongest pressure. Finally, "*Signing time*" is another enumeration having three values Total time, TimeUp ratio and Timedown ratio. Another attribute is stroke information which is identified when the pen velocity, pressure and angle decreases. These values are obtained in real time. These enumeration belongs to Feature selection. Similarly, there is one more enumeration class named "*Classification Method*" which supports different classifier techniques for real time signature verification such as SVM, DT and so on.

There are seven more stereotypes which extends metaclass Operations. "*ReadSignature*" stereotype is responsible for acquiring signature values such as x,y and pressure. The "*Normalization*" stereotype normalizes the signature size by scaling characters in horizontal and vertical direction. The values x_opoint and y_opoint represent the original points and x_tpoint, y_tpoint are the points after transformation. Stereotype "*Resampling*" is used to make raw data equidistant in time. Stereotype "*Addfeature*" is used to select the features and stereotype "*AddClassifier*" selects the classifier. Finally, stereotype "*Validation*" use to represent the different aspects of classification results.

4 Validation

The proposed UMLPDSV is validated through Internet Banking case study. The popularity of Internet Banking has increased many fold globally since last two decades. Leaving behind the traditional way of banking, the customers are encouraged to adopt this modern way of banking which not only save efforts for the customer but also save some basic expenditure on normal banking. Online Banking is an electronic payment system through which bank customers conduct a range of financial transactions. The type of transaction includes electronic bill payments, funds transfer, e-cheque payments between a customer and other account. So, personal identification is very important for the customers when using online banking and biometric systems are ways for providing efficient and reliable verification system to the users.

The proposed UMLPDSV is applied to the internet banking application where the application uses signature based authentication profile to verify the user. Figure 3 shows the internet banking application that uses real time dynamic signature verification.

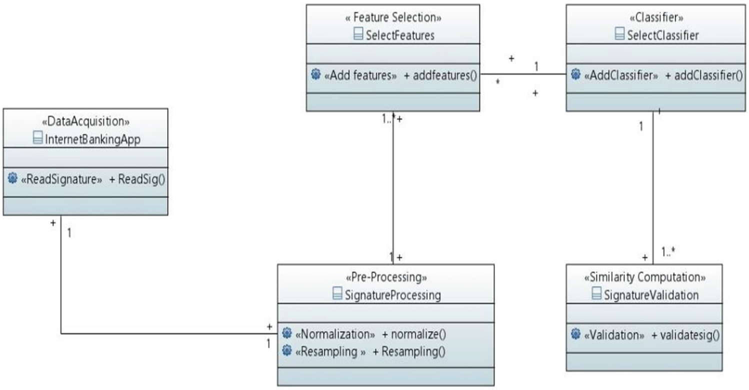


Fig. 3. UMLPDSV in internet banking application

The Internet Banking application captures the signatures from the GUI using the electronic device and pen, so the stereotype <<DataAcquisition>> is applied to acquire the signature values in real time in the form of coordinate stream. Also <<ReadSignature>> stereotype is applied to the operation "ReadSig()" that takes x, y and pressure values in real time. For signature processing, <<Pre-Processing>> stereotype is mapped to it. This stereotype has two operations "normalize()" and "Resampling". The stereotype "Normalization" is mapped to "normalize()" operation and "Resampling" stereotype is applied to "Resampling" operation. The <<featureSelection>> stereotype is applied to "selectfeatures" class. User can add multiple features. The "SelectClassifier" class is mapped to <<Classifier>> stereotype. The "addClassifier" operation is mapped to the <<AddClassifier>> stereotype. This stereotype is responsible for selecting the technique for classification to distinguish the real signatures from forged ones. Finally "Signature Validation" class uses <<Similarity Computation>> stereotype to indicate whether the signature is accepted or rejected. The operation "validatesig()" is mapped to the <<Validation>> stereotype.

5 Discussion

Real time dynamic signature verification is a complex process which is based on the time domain characteristics of the signature. In this research, we have proposed a UMLPDSV for real time dynamic signature verification. The proposed UMLPDSV is the first attempt towards the dynamic signature verification through model-driven approach. Consequently, UMLPDSV provides several benefits as compared to the other state-of-the-art approaches (e.g. [1, 4, 8, 10] etc.). The key advantages of UMLPDSV are: (1) it simplifies the specification/representation of requirements for dynamic signature verification process. (2) It provides strong basis to automatically transform the source models into different target models like Matlab and Python code to perform dynamic signature verification. Consequently, UMLPDSV significantly reduces the low level implementation complexity. (3) It provides foundation to integrate particular business process with dynamic signature verification. For example, it is fairly possible to extend UMLPDSV to incorporate banking applications requirements along with dynamic signature verification.

The work is still under progress and we are working on its enhancement as follows: (1) Currently, UMLPDSV doesn't provide any transformation capabilities to generate desired target models for signature verification. Consequently, we are working on transformation tool to automatically generate target models of several machine learning tools like Matlab code, Python, Weka, Rapidminer process etc. from the source UMLPDSV models. (2) At the moment, we apply UMLPDSV on relatively small case study. In this regard, we are working on different complex case studies to demonstrate the real application of UMLPDSV profile.

To summarize, the biggest benefit of UMLPDSV is the simplicity as the development and understanding of models is easy for different stakeholders because models neglect the implementation details that are not relevant to understand the logical behavior of a system. Furthermore, UMLPDSV models are too close to the problem domain and decrease the semantic gap amongst the ideas understandable by stakeholders and the language through which the solution is implemented. For example, UMLPDSV models can be automatically transformed to machine learning tools (e.g. RapidMiner process, Matlab code etc.). Therefore, user don't need to understand the complex low level implementation details regarding feature selection and classification for dynamic signature verification. Moreover, these models are easily extendible and used for workflow automation, document management and for electronic transactions in financial, healthcare, retail and other sectors.

6 Conclusions and Future Work

In this paper, the main idea is to resolve the challenges of real time dynamic signature verification mainly related to the signature variability patterns. In this context, a UMLPDSV (Unified Modeling Language Profile for Dynamic Signature Verification) has been proposed to specify the signature verification requirements at high abstraction level. Particularly, several stereotypes have been proposed to model dynamic signature verification requirements. It provides the basis to automatically generating target models of various machine learning tools from source UMLPDSV models. The proposed UMLPDSV is capable of acquiring signature data in real time. Furthermore, it is able to represent complete dynamic signature verification requirements in models that are understandable to all stakeholders, thus, providing a significant level of simplicity in signature verification process.

This is ongoing work and we are working on transformation engine for automatically generating target models of various machine learning tools such as Matlab code, Rapidminer process etc. from source models. Therefore, we intend develop a complete transformation tool and apply UMLPDSV on complex case studies in our next article.

References

1. Xia, X., Song, X., Luan, F., Zheng, J., Chen, Z., Ma, X.: Discriminative feature selection for on-line signature verification. Pattern Recogn. **74**, 422–433 (2018)
2. Vera-Rodriguez, R., Tolosana, R., Ortega-Garcia, J., Fierrez, J.: e-BioSign: stylus and finger-input multi-device database for dynamic signature recognition. In: Proceedings of 3rd International Workshop Biometrics Forensics (IWBF), March (2015)
3. Li, N., Liu, J., Li, Q., Luo, X., Duan, J.: Online signature verification based on biometric features. In: 49th Hawaii International Conference on System Sciences (2016)
4. Fernandes, J., Bhandarkar, N.: Enhanced online signature verification system. Int. J. Emerg. Trends Technol. Comput. Sci. (IJETTCS) **3**(6), November–December 2014
5. Shastry, A., Burchfield, R., Venkatesan, S.: Dynamic signature verification using embedded sensors. In: International Conference on Body Sensor Networks (2011)
6. Ibrahim, M.T., Kyan, M., Guan, L.: Online signature verification using global features. In: Electrical and Computer Engineering, CCECE 2009, Canadian Conference (2009)
7. Sonawane, R.C., Patil, M.E.: An Effective Stroke Feature Selection Method for Online Signature Verification. ICCCNT, Coimbatore (2012)
8. Chen, Z., Xia, X., Luan, F.: Automatic online signature verification based on dynamic function features. In: 7th IEEE International Conference on Software Engineering and Service Science (2016)
9. Rohilla, S., Sharma, A.: SVM based online signature verification technique using reference feature vector. In: Proceedings of the National Academy of Sciences, India (2016)
10. Kar, B., Mukherjee, A., Dutta, P.K.: Stroke point warping-based reference selection and verification of online signature. IEEE Trans. Instrum. Measur. **67**(1), 2–11 (2018)
11. Chaithra, D., Indira, K.: Handwritten online character recognition for single stroke Kannada characters. In: 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT) (2017)
12. Song, X., Xia, X., Luan, F.: Online signature verification based on stable features extracted dynamically. IEEE Trans. Syst. Man Cybern. Syst. **47**(10), 2663–2676 (2016)
13. Philip, J., Bharadi, V.A.: Signature verification SaaS implementation on Microsoft Azure Cloud. Procedia Comput. Sci. **79**, 410–418 (2016)
14. Tolosana, R., et al: Preprocessing and feature selection for improved sensor interoperability in online biometric signature verification, pp. 2169–3536. IEEE (2015)
15. Shin, J., Maruyama, K., Kim, C.M.: On-line signature verification using inter-stroke information. In: Proceedings of the International Conference on Research in Adaptive and Convergent Systems, pp. 80–84 (2016)

16. Wada, N., Hangai, S.: Online signature verification using gripping position and power. In: International Symposium on Intelligent Signal Processing and Communication Systems (2009)
17. Julita, A., Fauziyah, S., Azlina, O., Mardiana, B., Hazura, H., Zahariah, A.M.: Online signature verification system. In: 5th International Colloquium on Signal Processing & Its Applications (CSPA) (2009)
18. Alonso-Fernandez, F., Fierrez-Aguilar, J., del-Valle, F., Ortega-Garcia, J.: Online signature verification using tablet PC. In: Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis (2005)
19. Anwar, M.W., Rashid, M., Azam, F., Kashif, M.: Model-based design verification for embedded systems through SVOCL: an OCL extension for SystemVerilog. J. Des. Autom. Embed. Syst. **21**(1), 1–36 (2017). https://doi.org/10.1007/s10617-017-9182-z

Minimization of Power Pulsations in Traction Supply – Application of Ant Colony Algorithm

Barbara Kulesz(✉), Andrzej Sikora, and Adam Zielonka

Silesian University of Technology, ul. Akademicka 2, 44-100 Gliwice, Poland
barbara.kulesz@polsl.pl

Abstract. In this paper a particular problem related to transformation of AC voltage into DC voltage used in tram supply is considered. Variable component is always present in rectified voltage. Pulsation of rectified voltage is influenced by different factors. In 12-pulse system, where two secondary transformer windings are used (one delta-connected, the other star-connected), an additional factor increasing the pulsation is the unbalance of the output voltages at these windings. Tap changer may be used to compensate (equalize) those voltages and its setting is optimized here by applying ant colony algorithm. Different supply voltage variants have been considered, with particular attention paid to distorted voltage, containing 5th and 7th harmonic. The effects of applying ACO algorithm are demonstrated.

Keywords: Swarm optimization algorithm · Ant colony optimization
Traction voltage · Tram traction · Voltage transformation · Tap changer
Distorted voltage · Voltage unbalance · Multi-winding transformer

1 Swarm Intelligence Algorithm Used in Solving Optimization Problems

Demand for numerical optimization of issues arising in different branches of science and engineering has led to emergence of new, diversified algorithms, including metaheuristic ones. The heuristic approach is characterized by application of a practical method, which is not necessarily optimal, but it is sufficient, since it speeds up the process of finding a satisfactory enough solution.

Algorithms based on swarm intelligence are inspired by behaviour of large group of individual belonging to one species. This behaviour is co-ordinated even though central control is absent. An individual belonging to the group (swarm) is called a particle. Each individual is equipped with information on its current state (actions) and what was the previous state (actions) which awarded the best effects. Each individual is also able to inform other individuals (in some way particular to a given species) on what was the optimum way of action (which brought the best possible effects). Each individual is a sender and recipient of information. The system (swarm) evolves dynamically as a whole. Individuals which simulate this particular behaviour of other individuals which was most effective in the past, cause concentration of different individuals in some

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 399–411, 2018.
https://doi.org/10.1007/978-3-319-99972-2_33

regions of search areas; in this way a solution of the problem is reached [1]. Animal behaviour patterns such as communication, task allotment, decision making, localization of HQ, choice of partners, mating, propagation are mirrored in population algorithms.

Ant Colony Optimization [2, 14, 16, 17] is one of algorithms motivated by animal behaviour. This algorithm mimics social behaviour of ants, in particular localization of food source. Extensive list of other algorithms inspired by animal (including swarm) intelligence may be found e.g. in [3].

Population-based algorithms are steadily becoming more and more popular in electrical engineering. Different and diverse problems are nowadays solved, e.g. optimization of energy storage devices in a system containing wind power stations [4], placing of capacitor banks in distribution system [5], optimization of charging infrastructure in distribution system [6], estimation of three-phase asynchronous motor parameters [7], problems associated with railway transport [8], optimization of current balance in railway system [9]. In this paper we have applied ant colony algorithm (ACO) to the problem of "tuning" the output voltage of multi-phase traction rectifier transformer used in supply of DC tram traction lines.

Demand for numerical optimization of issues arising in different branches of science and engineering has led to emergence of new, diversified algorithms, including metaheuristic ones. The heuristic approach is characterized by application of a practical method, which is not necessarily optimal, but it is sufficient, since it speeds up the process of finding a satisfactory enough solution.

2 Rectification of AC Voltage in Tram Traction Substations in Poland

Rail vehicles such as locomotives, EMUs, tram cars and metro are supplied with DC voltage in Poland. The supply circuit consists of traction substation, traction network (catenaries and contact wires) and return feeders (see Fig. 1).

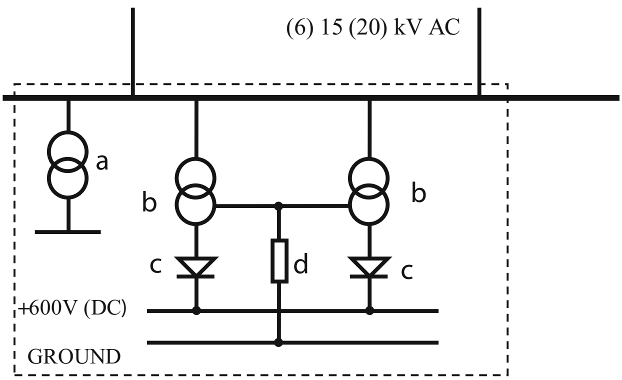


Fig. 1. Simplified scheme of tram traction substation: a – transformer supplying loads other than traction network, b – rectifier transformer, c – rectifiers, d – return cables

Traction substation consists of high voltage switching station, devices transforming ac three-phase energy into DC energy and DC switching station. Tram traction substations are supplied with 6, or 15 or 20 kV voltage.

Design of 12-pulse system is possible only when transformer's secondary side voltages are shifted in relation to each other by 30° (Fig. 2b). Such shift may be achieved by different means; a common solution is to use a three-winding transformer with two secondary windings, where one winding is delta-connected and the other is star-connected (Fig. 2a). Use of differently connected windings is due to the necessity of eliminating $3n$ magnetic flux harmonics and it is a standard rule applied in transformer design [10, 11]. Design of such transformer is relatively simple and transformer is practically batch-manufactured.

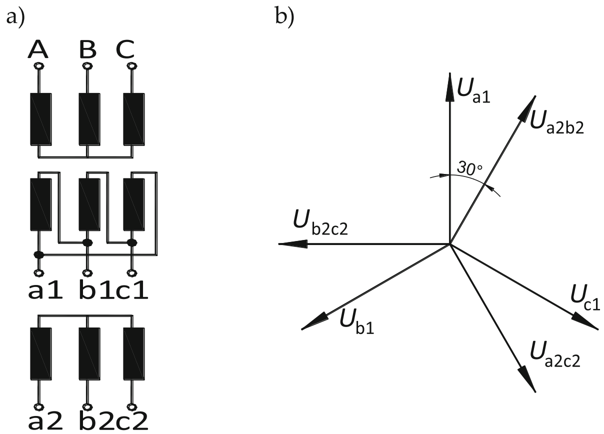


Fig. 2. Six-phase transformer winding configuration (a) and phase-to-phase voltage vectors (b)

3 Rectified Voltage Waveforms

Let us consider a theoretical case, where transformer's secondary side voltages are identical as to value and shifted in phase by 30°. When these voltages are rectified by means of uncontrolled diode rectifiers, then pulses in output voltage will be identical in shape (Fig. 3a). When we consider a more realistic case, when voltages start to differ in value, even though their distribution in space remains unchanged (the shift of voltages depends on spatial distribution of windings in transformer), then the output voltage waveform will show increased pulsations – see Fig. 3b.

If transformer's supply voltage is additionally distorted, then the increased pulsations in the rectified voltage will be further enhanced by the secondary side voltage unbalance. The pulsation magnitude will depend on supply voltage waveform shape.

Let us consider a case, when supply voltage contains additional 5th and 7th harmonic. We shall assume that RMS values of these harmonics are equal to 6% and 5% of RMS value of fundamental harmonic ($u_{5\ \%} = U_5/U_1 = 0.06$, $u_{7\ \%} = U_7/U_1 = 0.05$ – these are relative values, hence "%" in the subscripts). This gives an overall THD coefficient of supply voltage equal to 7.81%, which is less than mandatory limit of 8% prescribed by appropriate Polish regulations [12]. Let us further consider four following variants of harmonic phase shifts: a – $\varphi_5 = 0°$, $\varphi_7 = 0°$, b – $\varphi_5 = 0°$, $\varphi_7 = 180°$, c – $\varphi_5 = 180°$, $\varphi_7 = 0°$, d – $\varphi_5 = 180°$, $\varphi_7 = 180°$. When voltage with harmonic content is rectified, waveforms obtained display different shape and increased pulsations in relation to the sine wave supply – Fig. 4. Issues related to operation of rectifier transformers supplied with distorted voltage have been previously discussed e.g. in [13].

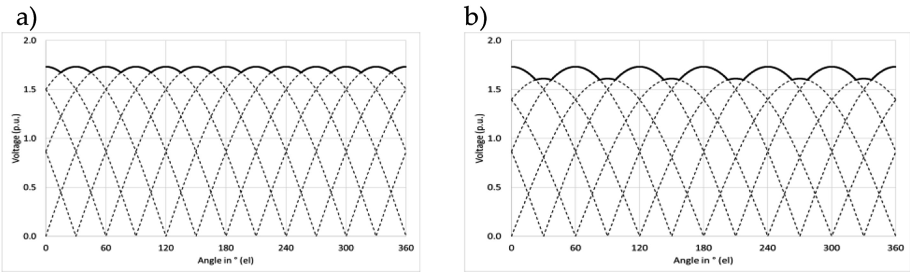


Fig. 3. Transformer voltage waveforms: secondary side phase-to-phase voltages are marked with dashed lines; rectified voltage is marked with thick continuous line (commutation impact has been neglected); (a) star - and delta winding voltages are identical, (b) star - and delta winding voltages differ ($U_D = 0.95U_Y$, where $U_{D,}$ U_Y – RMS-values of phase-to-phase delta and star winding voltages, respectively).

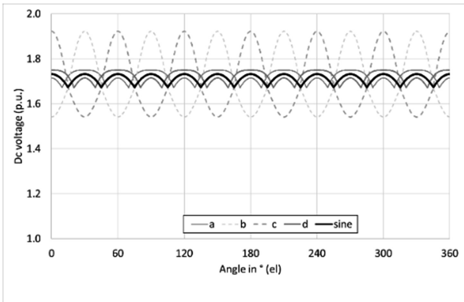


Fig. 4. Examples of rectified voltage waveforms: supply conditions are sine voltage wave (marked with thick black line) or voltage distorted by adding 5th and 7th harmonic in accordance with following variants: a – $\varphi_5 = 0°$, $\varphi_7 = 0°$, b – $\varphi_5 = 0°$, $\varphi_7 = 180°$, c – $\varphi_5 = 180°$, $\varphi_7 = 0°$, d – $\varphi_5 = 180°$, $\varphi_7 = 180°$, $u_5 = 6\%$, $u_7 = 5\%$. The windings are ideally balanced.

4 Optimization Problem

Let us first discuss the case of distorted supply in accordance with variant a ($U_5 = 0.06\ U_1$ and $U_7 = 0.05\ U_{1,}$ $\varphi_5 = 0°$, $\varphi_7 = 0°$). The DC waveform for this supply voltage is shown in Fig. 5.

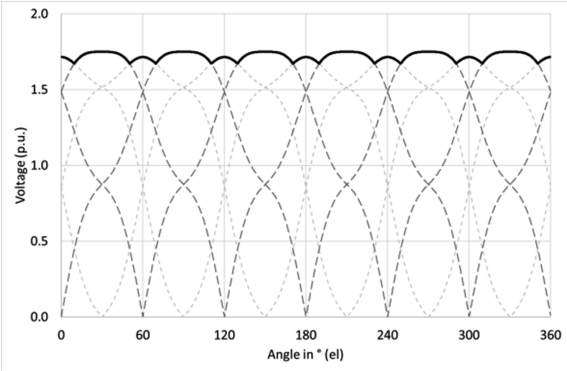


Fig. 5. Example of rectified voltage waveform: supply voltage is distorted by adding 5th and 7th harmonic: U_5 = 0.06 U_1 and U_7 = 0.05 U_1 (U_1, U_5, U_7 – RMS values of respective harmonics), harmonic phase angles are $\varphi_5 = 0°$, $\varphi_7 = 0°$. Phase-to-phase voltages of delta and star windings are marked with dashed lines and thick black line marks the output (DC) voltage

Now let us assume a slight disproportion in delta and star winding turns, e.g. difference equal to 8% (phase-to-phase induced voltage, fundamental harmonic). The rectified voltage obtained may be seen in Fig. 6.

There are different ways of defining the "quality" of rectified voltage (u_{DC}), one of them is by defining pulsation level, thus:

$$p = \frac{u_{\mathrm{DCmax}} - u_{\mathrm{DCmin}}}{2u_{DCav}}, \quad \text{where} \quad u_{DCav} = \frac{u_{\mathrm{DCmax}} + u_{\mathrm{DCmin}}}{2}. \tag{1}$$

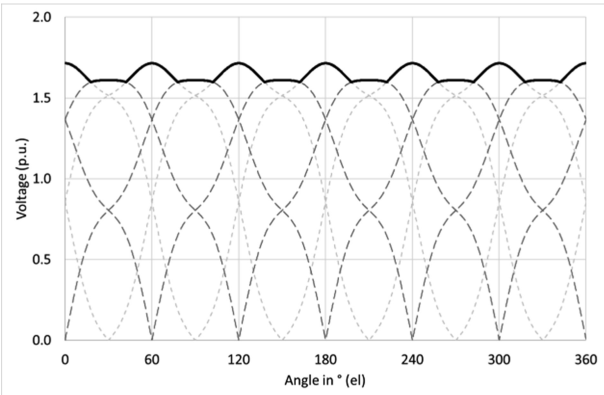


Fig. 6. Example of rectified voltage waveform: supply voltage is distorted by adding 5th and 7th harmonic: U_5 = 0.06 U_1 and U_7 = 0.05 U_1 (U_1, U_5, U_7 – RMS values of respective harmonics), overall THD = 7.81%, harmonic phase angles are $\varphi_5 = 0°$, $\varphi_7 = 0°$. Phase-to-phase voltages of delta and star windings are marked with dashed lines and thick black line marks the output (DC) voltage. Unbalance in star and delta winding turns (and voltages) is described in the text.

Pulsation calculated for different DC voltages waveforms is shown in Fig. 7. The curves are plotted against “voltage unbalance”, defined as the difference between phase-to-phase induced voltages of transformer’s secondary delta fand star windings. Unbalance coefficient equal to 1 corresponds to equality of delta and star voltages (in case of sinusoidal supply).

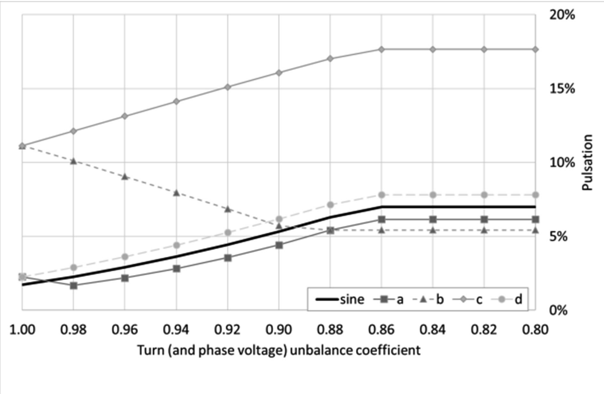


Fig. 7. Pulsation of DC voltage against turn and voltage unbalance of star- and delta-connected windings, for different supply voltage waveforms: sine means pure sinusoidal supply, variants a-d correspond to THD = 7.81% ($U_5 = 0.06\ U_1$ and $U_7 = 0.05\ U_1$, where U_1, U_5, U_7 – RMS values of fundamental, 5th and 7th harmonic) and differ by respective phase shifts of 5th and 7th harmonic (a – $\varphi_5 = 0°$, $\varphi_7 = 0°$, b – $\varphi_5 = 0°$, $\varphi_7 = 180°$, c – $\varphi_5 = 180°$, $\varphi_7 = 0°$, d – $\varphi_5 = 180°$, $\varphi_7 = 180°$).

Obviously for different waveforms of supply voltage, the influence of unbalanced turns (and, therefore, induced voltages in the delta and star secondary windings of the transformer) varies. If we assume that current in traction line is constant (which is justifiable), then pulsation in rectified voltage is reflected by pulsation in instantaneous power supplied to traction line.

Now, if we should compensate the inequality in the voltages by introducing the online tap changer to adjust voltage of particular winding, the resulting DC waveform could exhibit less ripple. The question is, how to calculate some “optimum” setting of the tap changer in view of varying harmonic content in the supply voltage, as defined by different voltage waveforms depending on THD and, more particularly, on the phase shifts of 5th and 7th harmonic? To extend the calculation range, we considered RMS-values of 5th and 7th harmonic and the same THD value (7.81%), but the phase angles of higher harmonics ($n = 5$ and $n = 7$) were changed continuously in the range of (0°; 360°).

We must note here that only the theoretical aspects of the problem have been analysed – we have not addressed a practical issue of how often the tap changer should be reset.

This issue is addressed with optimization procedure utilizing swarm population algorithm. The details are presented in subsequent section.

5 Optimization Procedure – Application of Ant Colony Algorithm

Marco Dorigo in his Ph.D. thesis [2] noted several ant behaviour patterns and he transferred these to algorithm solving travelling salesman problem. This approach inspired other uses of ant colony behaviour, for instance in the process of functional minimization; algorithm used here was first proposed by Taksari and used to minimize a function [14]. The crux of the problem is how to minimize the function, if we do not have any information on function properties; we can only determine values of this function at selected domain points:

$$f(x_1, x_2, \ldots, x_m) \to \min, \quad \text{where } (x_1, x_2, \ldots, x_m) \in \mathrm{D} \subset \mathrm{R}^m. \tag{2}$$

At introductory stage, i.e. initialization of algorithm, we start with determining algorithm parameters:

- population density N,
- maximum number of iterations I
- neighbourhood parameter α should be "large enough" at first, c. 60% of the variable range length.

Before iteration process starts, it is necessary to:

- generate randomly initial population of N ants x^i, $i = 1, 2, \ldots, N$; ants are located in the region of discussed problem $\mathrm{D} \subset \mathrm{R}^m$
- determine value of objective function (intensity of pheromone trail) for all N ants: $f(x_1^i, x_2^i, \ldots, x_m^i)$, for $i = 1, 2, \ldots, N$.
- from the ant population one with best location must be selected (with smallest value of objective function $x^{\text{best)}}$.

Basic part of algorithm:

- Starting with best-located ant (x^{best}), we generate a new ant population; these ants are randomly placed in the neighbourhood of best-located ant, i.e. with the most intensive pheromone trail:

$$x^i = x^{\text{best}} + \mathrm{d}x, \quad i = 1, 2, \ldots N, \tag{3}$$

  where $-\alpha \leq \mathrm{d}x_j \leq \alpha$, $j = 1,2,\ldots m$ is a value chosen at random.
- We choose a best located ant in this new population – i.e. we update x^{best}.
- Points 1–2 must be repeated I^2 times.
- We constrain the neighbourhood parameter $\alpha = 0.1 \cdot \alpha$
- Points 1–4 must be repeated I times.

The flow diagram of this algorithm has been described in [15].

The x^{best} obtained in this way is treated as the solution of the problem. It must be pointed out that the algorithm is metaheuristic, and therefore it should be run several more times (with identical input data) and the best result should be selected as the final solution.

In our problem, we first had to choose the objective function.

The function is total area under the pulse curves, calculated for one/sixth of supply voltage period, i.e. for 20/6 ms or 60° (el), frequency of line voltage is 50 Hz. The area, which depends on the waveform shape, will depend on the gain coefficients (corresponding to tap changer position or, more precisely, to voltage added or subtracted from particular winding by means of additional winding turns). These should be selected in such a way that sum of areas of two consecutive pulses is minimum. It must be noted that in this particular case the solution may be verified by a simple calculation.

In order to define objective function, let us first define the phase voltages of the fundamental harmonic of transformer's secondary voltage at one winding (delta-connected) (see Fig. 2b):

$$\begin{aligned} u_{a1\_1}(t) &= A_{ph\_1}\sin(\omega t) \\ u_{b1\_1}(t) &= A_{ph\_1}\sin(\omega t + \frac{2}{3}\pi) \\ u_{c1\_1}(t) &= A_{ph\_1}\sin(\omega t + \frac{4}{3}\pi), \end{aligned} \tag{4}$$

where A_{ph_1}, ph = $a1$, $b1$ or $c1$ is maximum value of phase voltage, fundamental harmonic.

Phase voltages of 5th harmonic are expressed as:

$$\begin{aligned} u_{a1\_5}(t) &= A_{ph\_5}\sin(5(\omega t + \varphi_5)) \\ u_{b1\_5}(t) &= A_{ph\_5}\sin\left(5(\omega t + \frac{2}{3}\pi + \varphi_5)\right) \\ u_{c1\_5}(t) &= A_{ph\_5}\sin\left(5(\omega t + \frac{4}{3}\pi + \varphi_5)\right), \end{aligned} \tag{5}$$

where A_{ph_5} is maximum value of phase voltage, harmonic $n = 5$.

Phase voltages of 7th harmonic are expressed as:

$$\begin{aligned} u_{a1\_7}(t) &= A_{ph\_7}\sin(7(\omega t + \varphi_7)) \\ u_{b1\_7}(t) &= A_{ph\_7}\sin\left(7(\omega t + \frac{2}{3}\pi + \varphi_7)\right) \\ u_{c1\_7}(t) &= A_{ph\_7}\sin\left(7(\omega t + \frac{4}{3}\pi + \varphi_7)\right), \end{aligned} \tag{6}$$

where A_{ph_7} is maximum value of phase voltage, harmonic $n = 7$.

Total phase voltages are expressed as:

$$\begin{aligned} u_{a1}(t) &= u_{a1\_1}(t) + u_{a1\_5}(t) + u_{a1\_7}(t) \\ u_{b1}(t) &= u_{b1\_5}(t) + u_{b1\_5}(t) + u_{b1\_7}(t) \\ u_{c1}(t) &= u_{c1\_1}(t) + u_{c1\_5}(t) + u_{c1\_7}(t) \end{aligned} \tag{7}$$

These are voltages at delta-connected winding; phase voltage is therefore equivalent to phase-to-phase voltage.

Other three phase voltages (i.e. voltages at the star-connected secondary winding) are obtained basing on phase-to-phase voltages of the first winding:

$$\begin{aligned} u_{a2}(t) &= k_a(u_{a1}(t) - u_{b1}(t))/\sqrt{3} \\ u_{b2}(t) &= k_b(u_{b1}(t) - u_{c1}(t))/\sqrt{3}\,. \\ u_{c2}(t) &= k_c(u_{c1}(t) - u_{a1}(t))/\sqrt{3}. \end{aligned} \tag{8}$$

Parameters k_a, k_b, k_c are directly related to correction of used number of turns of the second secondary winding. In other words, they describe the adjustment of taps.

We define the objective function as:

$$f(k_a, k_b, k_c) = \min \int_0^{T/6} (u_{ph}(t) - u_{ph\min})\mathrm{d}t, \tag{9}$$

where $u_{\mathrm{ph}}(t)$ is appropriate phase voltage for a given calculation interval (u_{a1} or u_{b1} or u_{c1} and so on) and u_{phmin} is minimum voltage value for the same time interval (*cf.* Fig. 8).

In the discussed case we have adopted the following parameters of ant colony algorithm: $N = 5$ (number of ants), $I = 6$ (number of iterations), and neighbourhood parameter $\alpha = 0.2$.

The gain coefficients k_a, k_b, k_c range from 0.85 to 1.15 ($k < 1$ – decrease and $k > 1$ – increase in number of turns, respectively).

We have tried running the algorithm with greater number of ants. The obtained results did not significantly differ from those achieved with $N = 5$ (and therefore we do not specify them here), but computation time was substantially longer. The calculations have been run on PC with Intel Core i5-2300, CPU 2.8 GHz. Computation time ranged from *c.*2100 to *c.*5800 s, depending on value of parameter N.

Numerical integration has been applied to calculus shown in Eq. (9), trapezoidal rule has been used, with number of nodes equal to 50. Integration interval is equal to 1/6 of fundamental harmonic period, since basic frequency of AC component in DC voltage is equal to $12f$ ($f = 50$ Hz) and two neighbouring pulses are compared (see Fig. 8).

Figure 8 shows the principle of the objective function. Areas marked as S1 and S2 are limited by the instantaneous value of delta- or star- phase voltage, respectively, and the average value of voltage ripple. Now, we take the integral of difference of these voltages: it is calculated as integral (Eq. 9) and then minimized. In this way the areas S1 and S2 are somewhat equalized.

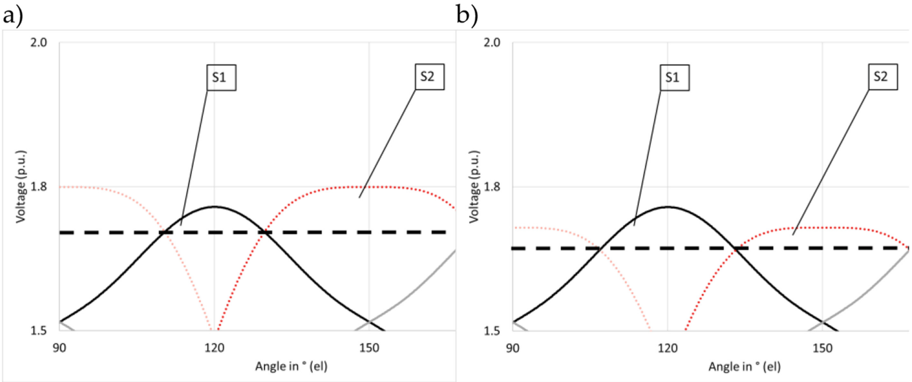


Fig. 8. Illustration of the compensation principle: (a) areas S1 and S2 under curves belonging to voltages of different windings (star- and delta-connected, respectively, marked here by continuous and dotted lines) differ; (b) tap-changer is applied, number of turns is slightly changed and so areas S1 and S2 are similar. Thick dashed line marks minimum of the voltage ripple.

The results of applying the algorithm are shown in Figs. 9 and 10. The ripple has been decreased (in accordance with proposed method) by 27% on the average.

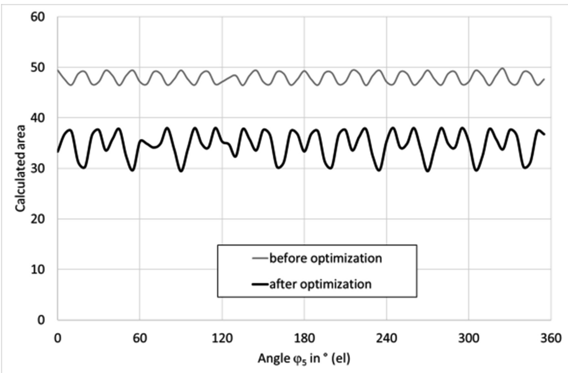


Fig. 9. Calculated area S1 + S2 (*cf.* Fig. 9) before and after optimization; area averaged over the entire range of phase shift of 7th harmonic of rectified voltage; before optimization (light-coloured line) and after optimization (dark-coloured line)

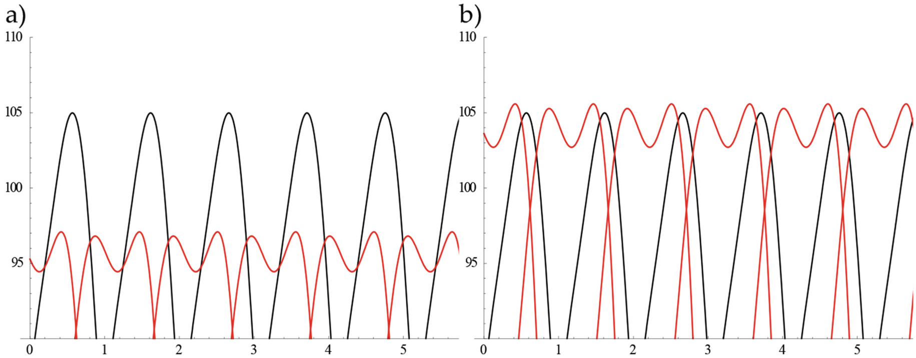


Fig. 10. Example of tap-changer correction of turn unbalance: phase-to-phase voltages of 6-phase transformer, black and red lines mark voltages of secondary delta- and star-connected winding, respectively; supply voltage is distorted by adding 5th and 7th harmonic: $U_5 = 0.06\ U_1$ and $U_7 = 0.05\ U_1$ (U_1, U_5, U_7 – RMS values of respective harmonics), overall THD = 7.81%, harmonic phase angles are $\varphi_5 = 5°$, $\varphi_7 = 110°$; (a) gain coefficient k_a, k_b, k_c is 1; (b) gain coefficient is corrected in accordance with ACO calculation results and is equal to 1.08747 (Color figure online)

6 Conclusion

Population algorithms (swarm intelligence algorithms) have been lately used for solving more or less complex technical problems. In the current paper we have considered a problem related to transformation of AC voltage into DC voltage used in tram supply (Fig. 1). On account of transformation circuit configuration (use of uncontrolled rectifiers and transformer with different winding connections), a variable (AC) component is always present in rectified voltage (Fig. 3). The magnitude of AC component is influenced by different factors such as configuration of rectifier circuit, shape of supply voltage waveform, load (Fig. 4). In 12-pulse system, where two secondary transformer windings are used (one delta-connected, the other star-connected), an additional factor increasing the pulsation is the unbalance of the output voltages at these windings (Fig. 6). We have demonstrated that in steady-state this AC component may be balanced by adjustment of voltage at one of these windings with the help of on-load tap changer. We investigated the case, when transformation circuit was supplied with distorted voltage, characterized by THD coefficient confined within limits set by appropriate legal regulations ($\leq 8\%$). Application of ant colony algorithm to the defined problem lets us determine the adjustment of tap changer at one secondary winding, and in this way the AC component between successive pulses becomes more balanced (in absolute values the difference may be as high as 27%) (Figs. 9 and 10).

References

1. Kennedy, J.: Particle swarm optimization. In: Encyclopedia of Machine Learning, pp. 760–766. Springer US (2011)
2. Dorigo, M.: Optimization, Learning and Natural Algorithms (in Italian). Ph.D. thesis, Politecnico di Milano, Italy (1992)
3. Karaboga, D., Gorkemli, B., Ozturk, C., Karaboga, N.: A comprehensive survey: artificial bee colony (ABC) algorithm and applications. Artif. Intell. Rev. **42**(1), 21–57 (2014)
4. Lee, T.-Y.: Operating schedule of battery energy storage system in a time-of-use rate industrial user with wind turbine generators: a multipass iteration particle swarm optimization approach. IEEE Trans. Energy Convers. **22**(3), 774–782 (2007)
5. Lee, C.-S., Ayala, H.V.H., Dos Santos Coelho, L.: Capacitor placement of distribution systems using particle swarm optimization approaches. Int. J. Electr. Power Energy Syst. **64**, 839–851 (2015)
6. Kumar, D., et al.: Reliability-constrained based optimal placement and sizing of multiple distributed generators in power distribution network using cat swarm optimization. Electr. Power Compon. Syst. **42**(2), 149–164 (2014)
7. Mohammadi, H.R., Akhavan, A.: Parameter estimation of three-phase induction motor using hybrid of genetic algorithm and particle swarm optimization. J. Eng. (2014)
8. Wu, Q., Cole, C., Mcsweeney, T.: Applications of particle swarm optimization in the railway domain. Int. J. Rail Transp. **4**(3), 167–190 (2016)
9. Zhu, H.P. et al.: A hybrid active power compensation device for current balance of electrical railway system. In: 2010 International Conference on Power System Technology (POWERCON), pp. 1–6. IEEE (2010)
10. McPherson, G., Laramore, R.D.: An Introduction to Electrical Machines and Transformers, 2nd edn. Wiley, New York (1990)
11. PN-EN 60076-1:2011 - wersja angielska: Transformatory. Wymagania ogólne. Polski Komitet Normalizacyjny, Warszawa. 75 p. [In Polish: Polish Standard. Transformers. General requirements. Warsaw: Polish Committee of Standardization]
12. Rozporządzenie Ministra Gospodarki z dn. 4 maja 2007r. w sprawie szczegółowych warunków funkcjonowania system elektroenergetycznego, Dziennik Ustaw 2007, nr 93, poz.623 [In Polish: Ministry of Economy ruling on detailed conditions of power engineering system operation, Journal of Laws 2007, No. 93, item 623]
13. Sikora, A., Kulesz, B., Grzenik, R.: Dwunastopulsowe i dwudziestoczteropulsowe układy przetwarzania napięcia przemiennego na napięcie stałe. Prace Naukowe Politechniki Śląskiej. Elektryka **3**, 29–64 (2015). [In Polish: 12-pulse and 24-pulse AC/DC voltage transformation circuits, Scientific Journal of Silesian University of Technology, Elektryka]
14. Toksari, M.D.: Ant colony optimization for finding the global minimum. Appl. Math. Comput. **176**(1), 308–316 (2006)
15. Sikora, A., Kulesz, B. Zielonka, A.: Application of swarm algorithm to solving voltage unbalance problem in DC tram traction supply system. In: 2018 Innovative Materials and Technologies in Electrical Engineering (i-MITEL), pp. 1–6, IEEE (2018). https://doi.org/10.1109/imitel.2018.8370483

16. Polap, D., Wozniak, M., Napoli, C., Tramontana, E., Damaševicius, R.: Is the colony of ants able to recognize graphic objects? In: Dregvaite, G., Damasevicius, R. (eds.) ICIST 2015. CCIS, vol. 538, pp. 376–387. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24770-0_33
17. Woźniak, M., Połap, D.: On some aspects of genetic and evolutionary methods for optimization purposes. Int. J. Electr. Telecommun. **61**(1), 7–16. doi:https://doi.org/10.1515/eletel-2015-0001

Model Driven Architecture Implementation Using Linked Data

Evgeny Cherkashin[1,2,5(✉)], Alexey Kopaygorodsky[3,5], Ljubica Kazi[4], Alexey Shigarov[1,2], and Viacheslav Paramonov[2]

[1] Irkutsk Scientific Center of SB RAS, 134 Lermontov Street, 664033 Irkutsk, Russia
[2] Matrosov Institute for System Dynamics and Control Theory of SB RAS, 134 Lermontov Street, 664033 Irkutsk, Russia
eugeneai@icc.ru
[3] Melentiev Energy Systems Institute of SB RAS, 130 Lermontov Street, 664033 Irkutsk, Russia
digger@istu.edu
[4] Technical faculty "Mihajlo Pupin", University of Novi Sad, Đure Đakovića bb, 23000 Zrenjanin, Serbia
ljubica.kazi@gmail.com
[5] National Research Irkutsk State Technical University, 83 Lermontov Street, 664074 Irkutsk, Russia

Abstract. We consider tools for developing information systems with use of Model Driven Architecture (MDA) and Linked Open Data technologies (LOD). The original idea of LOD is to allow the software designers to develop program systems integrated by means of common ontologies and web protocols. MDA Platform Independent Model (PIM) is expressed as set of UML diagrams. PIM forms a LOD graph and its namespace. All the PIM entities are defined as ontology resources, *i.e.* with URI references to LOD terms. This allows us to translate PIM UML model to a set of triples and store them in an ontology warehouse for further transformation into a Platform Specific Model (PSM). The ClioPatria ontology server and the SWI Prolog language are used as tools of PIM and PSM storage, querying and processing. The tools will allow us to mediate the MDA static means of code generation and configuration at development stage with the techniques of flexible data structure processing at run time, thus, producing even more productive information system development and maintenance techniques. This research corresponds to nowadays direction of Semantic Web Software Engineering.

Keywords: Model driven architecture · Linked open data
Logic programming · Knowledge based systems

1 Introduction

Model Driven Architecture (MDA) is a software development methodology based on transformations of models of the software under development. The input

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 412–423, 2018.
https://doi.org/10.1007/978-3-319-99972-2_34

models are to be more abstract than generated output ones, as well as main aim of MDA is technological support of special cases of software development techniques.

Linked Open Data (LOD) [1] technology has been suggested by W3C consortium to represent the semantic information in the published web content in a way that provides not only the possibility of its processing with software agents (Semantic Web), but also to link all available information into a single semantic graph using relations and global universal identifiers (URIs) of resources. The descriptive capabilities of semantic web technologies, HTML5 document publishing tools, and LOD technologies form an infrastructural basis for, *e.g.*, authoring and publishing documents [2]. The LOD provide a logical markup for the information presented in a document, informative basis for the different variants of visual representation and interpretation, logical connections with other documents, export information into other documents, procedural processing, *etc.* An important advantage of LOD usage in information environments is weakening of the requirements to the information warehouses: the document itself is a formalized data warehouse [3]. In some extent, this allows reallocation of the time spent on designing the database structure for storage of partially formalized documents to the process of solving a domain problem: the user (developer) markups the document text data with semantic meaning.

From the software designer point of view, LOD technologies allow development of program systems represent common and frequently used structures and data via common ontologies and web protocols [3]. An application of semantic models at run time [4] allowed software designers to partially cope with problem of requirements and data structure evolution in lifespan of information systems. A part of system configuration (static or dynamic) is built on domain ontology model interpretation, thus modifying information system behavior, resulting in a wider flexibility of system functioning. LOD data can be used for description of various contexts and aspects, interface elements, their behavior, user access regulation, transactions, and so on.

We distinct four popular approaches of implementing Platform independent model (PIM) to Platform specific model (PSM) transformation, taking into account the runtime and other implementation platform features. The first approach is model-to-model transformation with a model transformation language, such as QVT-X, ATL, *etc.* In this case PIM and resulting PSM are represented as XMI files. Model transformation language rules recognizes structures in PIM and construct structures of PSM, having account various `boolean` conditions calculated with helper rules. The second variant of transformation is realized with Domain Specific Language (DSL) infrastructure [5]. The transformation procedure is presented as translation of one DSL instance, *i.e.* PIM, to an instance in another language.

The third approach is a library-based one, it allows designing the resulting PSM and the source code with functions from a library that are applied to predicates or complex structures as subtasks. Usage of interpreted or ahead-of-time-compilation languages as Python, Perl, PHP, ASP.NET, JavaScript, HTML as a target language for PSMs decreases the development time and make it less complex. Thus, application of the libraries is the direct interaction of PIM as a superposition of applied function. The fourth approach is the multistage transformation of the source PIM [6] on the base of logical inference. The transformation is declared as a logical theory and a scenario of subgoals to be achieved.

The present research deals with developing software instrumental tools for general design of information systems using both Model Driven Architecture (MDA) and LOD [7]. MDA PIM is expressed as a set of UML diagrams (Class Diagram, State Diagram, *etc.*) logically connected with rules of transformations. PIM forms an RDF graph instance that can be assigned a namespace. All the PIM elements are defined as A-Box ontology resources, *i.e.* with URI references to LOD terms in a namespace of XMI standard. The graph is stored in an ontology warehouse for further transformation into a PSM. The ClioPatria ontology server and the SWI Prolog language [8] are used as tools of PIM and PSM storage, querying and processing.

The direct way of a transformation implementation is to use Prolog or a functional language, and we use Logtalk, a macro package of Prolog. The main advantage of the language usage is knowledge structuring thanks to its object-orientation. Objects are used for representation of the source PIM model and the resulting PIM structures as well as source code generators. Moreover, the transformation rules are easily integrated with libraries processing LOD and LOD warehouses.

The aim of this research is to construct tools for developing information systems with joint of MDA and LOD technologies in an indivisible cooperation.

2 Related Technologies and Standards

The most widely used technology of model transformation is ATL (ATLAS Transformation Language) [9] and its predecessor QVT, a OMG standard [10]. The language and its engine supports conversion from one XMI model to another one in the same format. The language structures describe recognition of properties of compositions of the source model and direct construction of new structures in the target model accounting the property values.

The ATL is supported with visual tools for rule construction. The tools are integrated in Eclipse IDE. Similar to ATL, Transformation Model Representation Language (TMRL) is developed in [11] but oriented on knowledge visual representation with following conversion in various production rules in CLIPS and OWL. The visual presentation of transforming rules are used also in [12], where a general approach is adopted for the purpose. In a recent paper [13], ATL is used to transform Computational Independent Model (CIM) represented as BPMN diagram into set of UML diagrams, defining PIM. Paper [14] proposes a model transformation CIM to PIM for web-applications, where CIM is represented with State and Use case UML-Diagrams logically connected with ATL rules. In [15] is proposed an evaluation of security aspects of distributed applications using MDA description; the evaluation is implemented a logical inference over the MDA synthesized logical security models. The paper [16] considers a highly decoupled distributed event-driven application environment, whose design and functioning is dynamically controlled by its three-level DSL description and MDA transformations, which support change propagation.

Paper [17] presents a practical approach with four-level ATL-description from HTML-page to a LOD structure (via stage of an MDA PIM). In [18], a converter of UML package structures represented in XMI was developed as a XSLT transformation. A wide overview of Western approaches to program transformation is given in [19,20], a transformation language PROMOL is suggested as well as its logical semantics. The book [19], together with John McCarthy's and Steve Russell's LISP technologies and Enn Tyugu's conceptual programming, covers the problem space history almost completely. OMG proposed ODM standard [21] for knowledge modeling. A Visual Ontology Modeler [22] is being developed for building and verifying OWL ontologies for ODM 1.0, as well as in the reverse direction, ontologies are represented as UML Class Diagrams [23].

An XML based specifications are used in representing metadata about web services, WSDL [24], and the WS-BPEL language [25] for describing orchestration of web services within execution of an instance of a business process. The idea of the language is to describe complex systems down to Web Services units, their internal and external structure and behavior, as well as services interaction in the script instance.

In our approach we use the structurally simplest language, Prolog, and its infrastructure, which is capable to load various model sources, including XMI, semantic graphs, *etc.* with corresponding import module, as well as their processing *em masse*, organizing multi-stage modular transformations. We also concentrate our efforts on programming in the small aspect of information system development, dealing with modeling and implementation software components composing software components.

3 Transformation Software Architecture

The architecture of the system is presented in Fig. 1. The PIM is designed with an UML editor as a set of diagrams. We use Modelio [26] as it contains means for formal definition of structured stereotypes, and it is open-sourced. The model of an information system is converted in a RDF graph encapsulated in a Logtalk object. The set of such objects are passed to the Transformation subsystem that is represented as a hierarchy of modules. The subsystem queries server storage of ontologies and library rules stored on the server via Pengines interface module. Library rules represent query helpers, which allow server-side preprocessing of the queries in its Inference machine.

The server is being implemented on the base of ClioPatria ontology storage, which in turn is realized with SWI-Prolog programming system with some C language subroutines. The ontology database supports durable storage and transactional querying as well as text search facility. ClioPatria SPARQL queries execution subsystem operates in a federated regime out of the box. The usage of

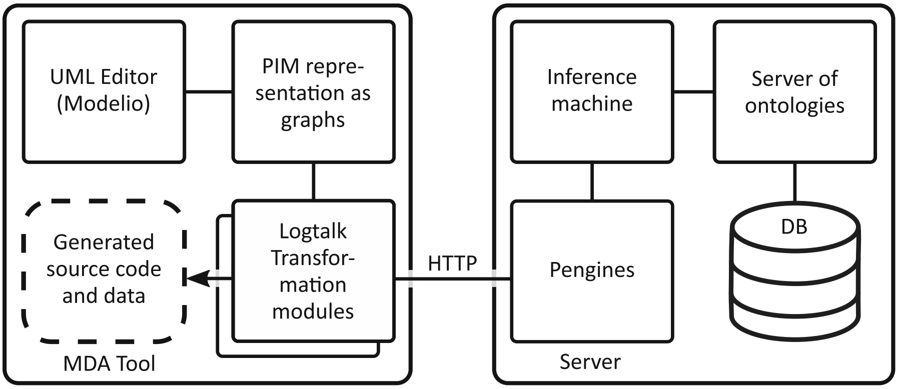


Fig. 1. A general architecture of the PIM transformation system

Prolog based infrastructure for triple processing significantly reduces the structural and semantic complexity of whole system. It supports several formats of compact storage of RDF data. The Pengines regulates access to the triples from clients, it is implemented as a simple protocol on top of HTTP.

Ontologies are loaded into ClioPatria via its web interface, or by means of startup script with a library predicates. ClioPatria has a simple interface for ontology browsing, transaction management, triple search by its literal values, and triple graph visualization for subset of triples. The system has tools for SPARQL query testing and debugging.

4 Implementation of Transformation

The transformation is implemented as a Logtalk library integrated with a RDF storage. All the data processing is under control of SWI-Prolog. No special DSL languages like ATL are needed, as we can do generalizations in LogTalk. PIM logical structure is defined as a set of sources: UML-diagrams, ontologies and data stored in the warehouse of ontologies as well as other databases. Logical connections of the elements of the sources are defined in a transformation program as well as the scenario of transformation. The output of the transformation is a set of configured interlinked Logtalk objects representing PSM. The source codes of the subsystems and the initial states of databases are generated from these object complexes.

At this stage of the research we implemented a processor of input XMI-2.0 files, which contain the UML-model of an information system, instances of transaction classes. The XMI files exported by Modelio UML designer consist of a model of the system as a set of top packages, and two profile packages. One profile package describes local definitions, and another represents the system and code generation configuration. The import processor converts their DOM trees into corresponding graphs of RDF triples. Each triple represents one relationship between two structural elements of an UML diagram. The instances have public interface used to query their graphs for the structural element compositions.

Platform model (PM) in a broad sense is the transformation procedure implemented as Logtalk components (instances) organized in an hierarchy of transformation modules (Fig. 2) in the image and likeness of our earlier work [6]. Each component (module) queries the sources and constructs two classes of data: own state of private data and structures of external objects. The states, in particular, contain type mappings between the source and target attributes, the implementation of designer decision on the general way of structure transformation.

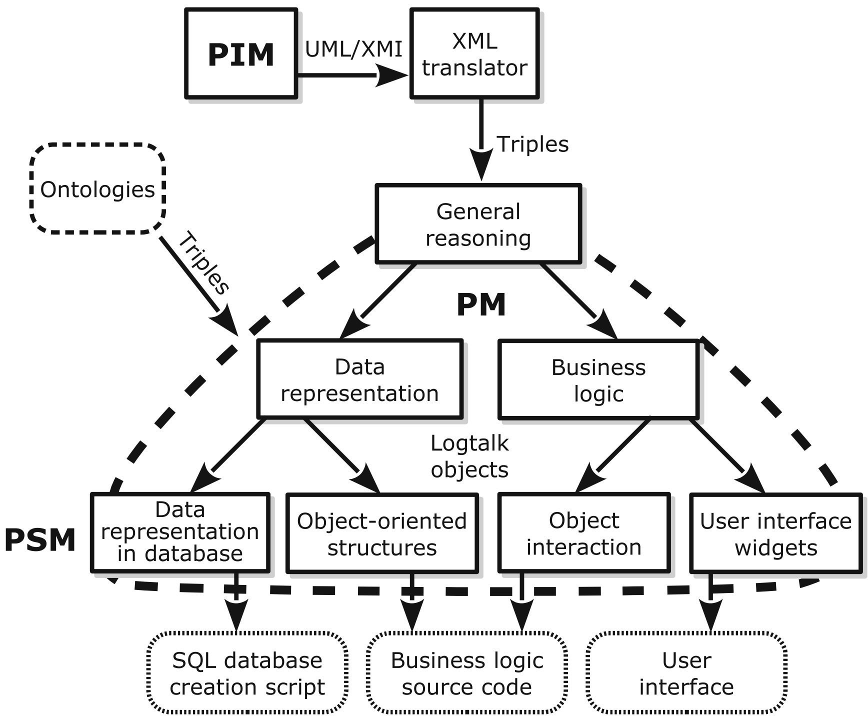


Fig. 2. A hierarchy of transformation modules representing PM

External objects are structures constructing PSM, *i.e.*, the set or a category of configured objects reflecting PIM structures and target sources and data. Transformation modules generate the objects as a results of general decisions. Their configurations are also inferred from the properties of structural elements of PIM and the profile configurations. The leaf nodes of the hierarchy is the scenario of PSM generation. The usage of pure logical approach instead of two languages allowed us to reduce the size and diversity of the PM description.

As soon as the scenario is fulfilled the PSM is converted into source codes and data. The conversion in [6] was made by means of partial application of text templates. In this case we do not use templates and generate the source code with simple procedures. SWI-Prolong module index contains a simple template engine module `simple-template` intended for generation of static HTML pages. The Prolog system also contain a dictionary-like structures and predicates for dictionary data manipulation, which can be used to represent data to be filled in the templates. So open-source SWI-Prolog infrastructure is well equipped for developing MDA instruments.

4.1 Inference of the Class Structure

Target class hierarchies in nowadays information system is diversified. In web-applications, a PIM class or structure is mapped to a table or a SQL-query, an input form, an JavaScript object at user's browser, it can be a part of data transfer protocol like a JSON or XML object. Some frameworks allow one to define structures and their relations and map them to rational databases, *e.g.*, Django [5] or Entity framework. Most of the CASE-subsystems in UML composition tools do not support export of Class Diagram into ORM (Object-Relational Mapping) structures.

The main difference of MDA with respect to CASE technologies is the possibility of adaptation of PM to the special way of software development used in a particular software development group. To put it on another way, MDA implements the standard transformations, and there is tools for modification of the standard procedures to implement own special techniques. For example, if a programmer wants to define a storage class for structures, the corresponding modification of standard transformation must be carried out. Similarly a whole UML class hierarchy can be immersed into an ORM by means of a programmer-defined stereotype and its implementation. Even an ORM itself may be implemented by means of a special transformation.

4.2 Inference of the Properties

One of the interesting application of the Semantic WEB technologies, namely Linked Open Data, is the inference of the properties of the generated structural elements. The properties of PSM elements, *e.g.*, instance attributes, are derived from PIM UML structure elements, *i.e.*, their standard UML definitions, analysis of relations being involved elements, features of combinations of associated stereotypes and tag values, as well as knowledge and data obtained from local ontology warehouse and SPARQL queries to other servers (cached for a better performance), for example, to `DBPedia.org`.

Consider the user interface template generation for a web or desktop application. The model of the interface has a description of the structure and a specification of the properties of the constituent widgets. If we refer from PIM with tag values `rdf:object` and `rdf:type` of a special stereotype `«DBPedia»` to a corresponding DBPedia resource we can use the DBPedia subgraph for constructing `title` and `placeholder` attributes of an input field. In principle, we can figure out a label for user's locale. Some UML-structures (OCL) and relations may be interpreted constructively as methods of objects: events, handlers, subscribers, database triggers and selectors (lookup widgets).

Consider some transformation rules. In order to import PIM we need to create holding instance `sample` and initialize it. In this a very simple example we represent a class `Person` in a Class diagram, server access code is omitted.

```
:- object(sample,instantiates(packageclass)).
:- initialization((::load_file('sample.xmi'),::reg_prefixes,::process)).
:- end_object.
```

At the import stage, the source XMI converted into a graph. We show its small part in the turtle format.

```
@base <File://.../samples.xmi#>.
<#__k9i9BN_Eeijsfq5>
  a uml:Class;
  <#ownedAttribute> <#RN_Eeijsfq5>, <#hN_Eeijsfq5>, <#9xN_Eeijsfq5>;
  <#ownedOperation> <#BN_Eeijsfq5>; rdfs:label "Person".
<#__k9i9RN_Eeijsfq5>
  a uml:Property; <#type> <http://...#String>; <#visibility> "public";
  rdfs:label "Name".
<#BN_Eeijsfq5>
  a uml:Operation ;
  <#ownedParameter> <#hN_Eesfq5>, <#hN_Eesfq5>; <#visibility> "public";
  rdfs:label "Pay" .
```

Construction of the target class is realized by querying the graph with a parametric object `query(_Graph)`.

```
% ?- direct(sample,lp,cp)::tr(class,person,ID).
:- object(direct(_Package, _Local, _Code)).
tr(class, Class, ClassID):- ::package(Package), % obtain a PIM
  query(Package)::class(Name, ClassID),
  ::new(Class, [instantiates(class)]),
  ::new(Attributes,[instantiates(params)]),
  ::new(Methods, [instantiates(methodlist)]),
  Class::name(Name), % get class name.
  forall(::tr(attribute, Attribute, ClassID, _), % All attributes
      Attributes::append(Attribute)),
  forall(::tr(method, Method, ClassID,_), % All methods
      Methods::append(Method)),
  Class::attributes(Attributes), Class::methods(Methods). % construct
:- end_object.
:- object(query(_Graph)). % querying object
class(N, ID):- ::xmi(X), % the parameter N=Name, X=XMI
  X::rdf(ID, rdf:type, uml:'Class'), X::rdf(ID, rdfs:label, literal(N)).
:- end_object.
```

Target structures constructed by blocks of elements and other blocks like in `llvmlite` library. New `items` are appended or prepended in a block. Rendering the sources is realized with methods of construction classes.

```
:- object(class, specializes(code_block)).
renderitem(Item, Result):- ^^renderitem(Item, Result). % call parent.
render(Result):- ^^render(Name), .....
 root::indent, % Python block indentation
 (  ::item(attributes(Attributes))-> Attributes::render(DAttrs),
       root::iswritef(ConstructorDef, 'def __init__(self, %w):',[DAttrs]),
    root::indent, Attributes::items(InstAttrs),
    findall(S, (lists::member(Attr, InstAttrs), Attr::item(name(AttrName)),
                root::iswritef(S, "self.%w=%w", [AttrName, AttrName])),
            AttrAssigns), root::unindent,
    AttrList=[ConstructorDef|AttrAssigns];
    root::iswritef(ConstructorDef, 'def __init__(self):', []),
    root::indent, root::iswritef(Pass,'pass', []),
    root::unindent, AttrList=[ConstructorDef, Pass]    ),
   .....
  lists::append(AttrList, Methods, StringList), root::unindent,
  Result=[Signature | StringList]. % Join source lines.
:- end_object.
```

As a result we obtain the following Python code.

```
# ?- person::render_to(write).
class Person:
    def __init__(self, Name, Family, BirthDay):
        self.Name=Name
        self.Family=Family
        self.BirthDay=BirthDay
    def Pay(self, Amount):
        pass
```

5 Applications

The described MDA tools are used in a number of interesting projects. The first project was to synthesize a Python interface driver of a cash register for a ticket system in a private cinema.

In the only Russian commercial planetary, a problem of connecting new cash register to a *ad hoc* ticket and billing system arose. The resources were limited by two weeks for main development and one programmer. There was an old version of a program for the previous cash machine, a documentation to the new one, as well as test utility implementing the protocol obtained from vendor driver pack. The description of the protocol in the manual was not precise: the sense of some fields were not understandable, the description of the automatons modeling the behavior in communications was too abstract, the documentation described two versions of the protocol, programmer has no experience related cash machines.

The decision of application of the MDA was made after the first investigation of the situation. The protocol commands and the automatons were represented as UML Class and State diagrams. After the application of the transformation adapted modules an interface component was generated. The component encoded API call of a command into the bytes of protocol string and decoded the answer byte string from the cash machine into the answer object. The interaction automaton of the computer and the cash device converted to a state machine with special states denoting exceptions related to undocumented and faulty behavior. Tree levels of the interaction has been implemented: low-level protocol for message exchange, middle-level automaton cash machine state control and computer synchronization, and high-level algorithm implementing cash machine usage in fiscal operations within internet ticket booking system.

The low- and middle-level interaction were tested on dumped data of the test tool. The tests showed the protocol documentation mistakes and version variant. It also helped in PM and PIM refinement. The first two levels of interaction started functioning in two weeks. Another reserved two weeks was devoted to refinement of the fiscal operation, ticked printing design and deployment.

The second active project is a technological support of the development of New Generation Sequencing computation infrastructure for investigation of lake Baikal microbiome. The investigation is a composition of elementary bioinformatic operations implemented by external software like Mothur and R. The developed tools are used to describe the operations as elements of a dataflow graph [27]. Each operation communicates other ones with connections of ports. The internal structures of its ports and parameters are represented with classes of UML Class Diagram. The PM adopted to generate set of stub Java classes reflecting Mothur operations of Rapidminer Studio. LOD is used to supply user descriptions in user interface for the operations.

6 Conclusion

We considered an implementation and applications of usage of Model Driven Architecture and Linked Open Data, a Semantic Web, technologies in developing software [28] on the base of model transformation. The transformation is realized as a hierarchy of objects represented in LogTalk. Results of analysis of the source model and synthesis of the target one are obtained with logical inference.

Usage of logical programming language simplifies the transformation procedure programming, the programmer is allowed to express semantics of recognized structures as properties of objects, manipulate object properties further. It also allows us to integrate the transformations with other Prolog libraries and services without usage of special language structures. The transformation logically connect elements of source objects: UML diagrams, RDF A- and T-boxes, LogTalk modules, which are just structures to be processed accounting to a scenario.

The approach is being tested in software development processes for synthesis of business-objects of information systems carcasses, interface modules for a protocol, as an ontology development tool, *etc.* Further development of the software is aimed at extending transformation modules, creating libraries of platform descriptions for open-source frameworks, develop techniques of change propagation, and developing open-source UML editors to support UML 2.5 profile specification [29] and XMI export.

Acknowledgment. The results are obtained with the partial support of the various projects: Irkutsk scientific center of SB RAS No 4.2; The Council for grants of the President of Russian Federation, state support of leading scientific schools of the Russian Federation (NSH-8081.2016.9); Russian Foundation for Basic Research, grants 17-07-01341, 18-07-00758 and 17-47-380007. The results obtained with the use of the network infrastructure of Telecommunication center of collective use “Integrated information-computational network of Irkutsk scientific-educational complex” (http://net.icc.ru). The authors are grateful to the community of Linked Open Vocabularies (http://lov.okfn.org/dataset/lov/) resource for assistance in the search for domain ontologies and `github.com` for hosting sources at https://github.com/isu-enterprise/icc.xmitransform.

References

1. Bizer, C., Heath, T., Berners-Lee, T.: Linked data - the story so far. Semant. Web Inf. Syst. **5**(3), 1–22 (2009)
2. Capadisli, S., Guy, A., Verborgh, R., Lange, C., Auer, S., Berners-Lee, T.: Decentralised authoring, annotations and notifications for a read-write web with dokieli. In: Cabot, J., De Virgilio, R., Torlone, R. (eds.) ICWE 2017. LNCS, vol. 10360, pp. 469–481. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-60131-1_33
3. Cherkashin, E., Orlova, I.: Instrumental tools for construction of the digital archives of the documents based on Linked Data. Modern Technol. Syst. Anal. Model. **4**(56), 100–107 (2017). https://doi.org/10.26731/1813-9108.2017.4(56).100-107. http://stsam.irgups.ru/sites/default/files/articlespdffiles/100-107.pdf. (in Russian)
4. Kopaygorodsky, A.: Use of ontologies in semantic information systems. Ontol. Des. **4**(14), 78–89 (2014). (in Russian)
5. Annenkov, D., Cherkashin, E.: Generation technique for Django MVC web framework using the stratego transformation language. In: Proceedings of 36th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 20–24 May, Opatija, Croatia, pp. 1084–1087 (2013)
6. Cherkashin, E., Larionov, A., et al.: Logical programming and data mining as engine for MDA model transformation implementation. In: Proceedings of 36th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 20–24 May 2013, Opatija, Croatia, pp. 1029–1036 (2013)
7. Frankel, D.: Model Driven Architecture: Applying MDA to Enterprise Computing, 1st edn. Wiley, New York (2003). 352 p
8. Wielemaker, J., Beek, W., Hildebrand, M., Ossenbruggen, J.: ClioPatria: A SWI-prolog infrastructure for the semantic web. Semant. Web **7**(5), 529–541 (2016). https://doi.org/10.3233/SW-150191
9. Jouault, F., Allilaire, F., Bezivin, J., Kurtev, I.: ATL: a model transformation tool. Sci. Comput. Program. **72**(1–2), 31–39 (2008)
10. The MOF Query/View/Transformation Specification Version 1.1. http://www.omg.org/spec/QVT/1.1
11. Berman, A., Grishchenko, M., Dorodnykh, N., Nikolaychuk, O., Yurin, A.: A model-driven approach and a tool to support creation of rule-based expert systems for industrial safety expertise. In: Proceedings of the 12th International Forum on Knowledge Asset Dynamics (IFKAD-2017) - Russia, Graduate School of 16 Management of St. Petersburg University, St. Petersburg, pp. 2034–2050 (2017)

12. Belghiat, A., Bourahla, M.: UML class diagrams to OWL ontologies: a graph transformation based approach. Int. J. Comput. Appl. **41**, 41–46. https://doi.org/10.5120/5525-7566
13. Rhazali, Y., Hadi, Y., Mouloudi, A.: Model transformation with ATL into MDA from CIM to PIM structured through MVC. Procedia Comput. Sci. **83**, 1096–1101 (2016). https://doi.org/10.1016/j.procs.2016.04.229
14. Rhazali, Y., Hadi, Y., Chana, I., Lahmer, M., Rhattoy, A.: A model transformation in model driven architecture from business model to web model. IAENG Int. J. Comput. Sci. **45**(1), 104–117 (2018). http://www.iaeng.org/IJCS/issuesv45/issue1/IJCS45116.pdf
15. Hamid, B., Weber, D.: Engineering secure systems: models, patterns and empirical validation. Comput. Secur. **77**, 315–348 (2018). https://doi.org/10.1016/j.cose.2018.03.016. https://www.sciencedirect.com/science/article/pii/S0167404818303043
16. Tragatschnig, S., Stevanetic, S., Zdun, U.: Supporting the evolution of event-driven service-oriented architectures using change patterns. Inf. Softw. Technol. **100**, 133–146 (2018). https://doi.org/10.1016/j.infsof.2018.04.005. https://www.sciencedirect.com/science/article/abs/pii/S0950584916303251
17. Bouougada, B., Bouchiha, D., Malki, M.: A framework for reengineering web applications into linked data based on MDA. In: ACM International Conference Proceeding Series, 23–25 November 2015. https://doi.org/10.1145/2816839.2816880
18. UMLtoOWL: Converter from UML to OWL. http://www.sfu.ca/~dgasevic/projects/UMLtoOWL/
19. Štuikys, V., Damaševičius, R.: Meta-program development as a model transformation process (2013). https://doi.org/10.1007/978-1-4471-4126-6_11
20. Štuikys, V., Damaševičius, R., Targamadze, A.: A model-driven view to meta-program development process. Inf. Technol. Control **39**(2), 89–99 (2010). http://itc.ktu.lt/index.php/ITC/article/download/12302/6838
21. Odm, UML profile for OWL. http://www.omg.org/spec/ODM/1.0/PDF/
22. Ontology Domain Modeling example. https://thematix.com/tools/vom/
23. Owl, UML Visualizer. http://owlgred.lumii.lv/
24. Web Services Description Language - Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Web_Services_Description_Language
25. Business Process Execution Language - Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Business_Process_Execution_Language
26. Modelio Open Source - UML and BPMN free modeling tool. https://www.modelio.org/
27. Johnston, W.M., Hanna, J.R.P., Millar, R.J.: Advances in dataflow programming languages. ACM Comput. Surv. **36**, 1–34 (2004)
28. Semantic WEB Software Engineering. http://www.webist.org/Documents/Previous_Invited_Speakers/2012/WEBIST2012_Pan.pdf
29. Unified Modeling Language, ver. 2.5 standard description. http://www.omg.org/spec/UML/2.5/PDF

Relationship Between Cohesion and Coupling Metrics for Object-Oriented Systems

Samuel António Miquirice(✉) and Raul Sidnei Wazlawick

Universidade Federal de Santa Catarina, Florianópolis, Brazil
samuel.miquirice@gmail.com, raul@inf.ufsc.br

Abstract. Cohesion and coupling are regarded as fundamental features of the internal quality of object-oriented systems (OOS). Analyzing the relationships between cohesion and coupling metrics plays a significant role to develop efficient techniques for determining the external quality of an object-oriented system. Researchers have proposed several metrics to find cohesion and coupling in object-oriented systems. However, few of them have proposed an analysis of the relationship between cohesion and coupling. This paper empirically investigates the relationships among several cohesion and coupling metrics in object-oriented systems. This work attempts to find mutual relationships between those metrics by statistically analyzing the results of experiments. Three open-source Java systems were used for experimentation. The empirical study shows that cohesion and coupling metrics are inversely correlated.

Keywords: Cohesion metrics · Coupling metrics
Object-oriented system

1 Introduction

Cohesion and coupling are regarded as fundamental features of the internal quality of object-oriented system (OOS). Class cohesion, according to [3] can be defined as being the degree of relationship between the members (attributes and methods) of a class. Class coupling, on the other hand, can be defined as the degree in which a class is related to other classes.

To ease the evolution of object-oriented systems, good designers should maximize cohesion (because cohesion promotes encapsulation) and minimize coupling (because coupling hinders encapsulation).

Researchers have proposed several metrics to find cohesion and coupling in object-oriented systems. However, few of them have proposed an analysis of the relationship between cohesion and coupling.

Studying the relation between internal quality attributes helps explaining some practical issues and results of the relation between internal and external

Supported by organization x.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 424–436, 2018.
https://doi.org/10.1007/978-3-319-99972-2_35

quality attributes. This study may help software engineers and practitioners to figure out the quality factors that must be considered and focused on during the quality assessment process [3].

This paper empirically investigates the relationships among several cohesion and coupling metrics in object-oriented systems. The study involves three Java open-source systems.

In this paper, fourteen cohesion metrics are considered: CBMC [15], ICBMC [33], OL_n [34], PCCC [1], LCOM1 [16], LCOM2 [17], LCOM3 [26], LCOM4 [23], LCOM5 [22], TCC [10], LCC [10], LSCC [4], SCOM [18], and CC [12]. Additionally, eight coupling metrics are considered: CBO [16], CBO_U [24], CBO_IUB [24], RFC [16], MPC [27], DAC [27], DAC2 [27], and OCMEC [13]. We selected those cohesion and coupling metrics because they are widely theoretically and empirically studied by different researchers.

Results show that the selected metrics are negatively correlated. It means that cohesion and coupling metrics are inversely correlated.

The paper is organized in four additional sections. Section 2 presents related works including an overview of the selected cohesion and coupling metrics. Section 3 describes the considered systems and the data collection process. Section 4 reports the correlation study analysis and discusses its results. Section 5 present conclusions and future work. Appendixes B and A respectively present the definition of the cohesion and coupling metrics used in this study.

2 Related Works

Cohesion and coupling are internal quality attributes. They can be measured by means of the source code only, whereas the external quality attributes.

In the software engineering community, some researchers intuitively believe that, high cohesion is related to low coupling and vice versa. However, the relationship between the cohesion and coupling metrics was not sufficiently studied to support this intuition. Research carried out in the analysis of the relationship between cohesion and coupling metrics did not cover a representative number of metrics to guarantee the generalization of results.

Researchers have proposed several metrics to find cohesion and coupling in object-oriented systems. However, few of them have proposed an analysis of the relationship between cohesion and coupling. Mateos *et al.* [28] and Anabalon *et al.* [5] report the research they performed to analyze the correlation between cohesion and coupling metrics (CBO, RFC, and LCOM) with complexity metrics (DW, DMR, ME, and MRS). On the other hand, Kumar *et al.* [25] analyzed the correlation between cohesion and coupling metrics (CBM, IC, CAM, LCOM3, CE, CA, LCOM, RFC, and CBO) with fault proneness. However, the focus of the present work is on the analysis of the correlation between cohesion and coupling metrics.

In [2,3,9], the authors analyzed the correlation between cohesion and coupling metrics, but conclusions were hard to generalize because a small number of metrics were used.

Linda Brad *et al.* in their research use four cohesion metrics and only one coupling metric [9]. Al Dallal in her research used five cohesion metrics and eight coupling metrics [2], two years later in a new research she used four cohesion metrics and eight coupling metrics [3].

This work analyzes fourteen cohesion metrics (LCOM1, LCOM2, LCOM3, LCOM4, LCOM5, CBMC, ICBMC, OL_n, PCCC, LSCC, CC, SCOM, TCC, and LCC) and eight coupling metrics (CBO_IUB, CBO_U, CBO, RFC, MPC, DAC, DAC2, and OCMEC).

The LCOM1 metric counts the number of pairs of methods that do not share instance variables [16]. Chidamber and Kemerer [17] propose another version of the LCOM metric (LCOM2), which calculates the difference between the number of pairs that do and do not share instance variables.

Li and Henry [26] use an undirected graph that represents each method as a node and the sharing of at least one instance variable as an edge. They define LCOM3 as the number of connected components in the graph. The graph is extended in [23] by adding an edge between a pair of methods if one of them invokes the other.

Hitz and Montazeri [23] introduce connectivity metrics to apply when the graph has one component (LCOM4). Henderson-Sellers [22] propose a LCOM5 metric that considers the number of methods referencing each attribute.

Fernández and Peña [18] propose a class cohesion metric called Sensitive Class Cohesion Metric (SCOM), which considers the cardinality of the intersection between each pair of methods.

In the Class Cohesion (CC) metric proposed in [12], the degree of similarity between methods is used as a basis to measure class cohesion. The similarity between a pair of methods is defined as the ratio of the number of shared attributes to the number of distinct attributes referenced by both methods. Cohesion is defined as the ratio of the sum of the similarities between all pairs of methods to the total number of possible pairs of methods.

In [4], the authors proposed a different formula for the similarity between a pair of methods, calculated as the ratio of the number of shared attributes to the total number of attributes in the class. They defined their Low-level design Sensitive Class Cohesion metric (LSCC), to be the relative similarities among the pairs of methods in the class. Chae et al. [15] proposed Cohesion Based on Member Connectivity (CBMC) to be the connectivity factor of its reference graph, scaled by the structure factor of its reference graph;

Xu and Zhou [33] proposed an Improved Cohesion Based on Member Connectivity (ICBMC) as a way to improve the CBMC metric. Yang [34] proposed OL_n, as the common strength of attributes.

Al Dallal [1] proposed Path Connectivity Class Cohesion (PCCC) for the connectivity pattern of the class reference graph, as the ratio of the number of simple paths in reference graph G to the number of simple paths in the corresponding graph FG. To evaluate class coupling in [24] it was proposed CBO_IUB, which counts the number of classes, excluding the inherited classes that use the attributes or methods of a class of interest.

CBO_U [24] counts the number of classes, excluding the inherited classes that are used by the methods of the class of interest. Coupling Between Objects (CBO) [16] can be calculated as CBO_U + CBO_IUB.

Response For a Class (RFC) [16] is measured by counting the number of methods in the class of interest and the number of distinct methods of the other classes directly invoked by the methods of the class of interest.

Message passing coupling (MPC) [27] is measured by counting the number of method invocations in the class of interest.

Data abstraction coupling (DAC) [27] counts the number of attributes in a class of interest whose types are of other classes, while DAC2 [27] count the number of distinct classes used as types of the attributes of the class of interest.

The OCMEC metric [13], counts the number of distinct classes used as types of the parameters of the methods.

Appendices B and A give the definitions of the fourteen cohesion metrics (LCOM1, LCOM2, LCOM3, LCOM4, LCOM5, CBMC, ICBMC, OL_n, PCCC, LSCC, CC, SCOM, TCC, LCC) and eight coupling metrics (CBO_IUB, CBO_U, CBO, RFC, MPC, DAC, DAC2, OCMEC).

3 Selected Systems

We collected some data in [2,3] where a Java tool was developed by Al Dallal [3] and the Borland Together Tool was used to automate the extraction of cohesion and coupling metrics. The data was obtained in three Java open-source systems from different domains: Art of Illusion, JabRef, and FreeMind, all this being open source systems randomly selected from SourceForge.

The choice of the three systems was motivated by their application in some related works [2,3], which facilitated the comparison of the results obtained.

Art of Illusion, JabRef, and FreeMind consist of 430, 306 and 363 concrete classes (not abstract classes or interfaces), respectively. The descriptive statistics for each of the selected cohesion and coupling metrics including the minimum, 25% quartile, mean, median, 75% quartile, maximum value, and standard deviation (adapted from [2,3]), is shown Tables 1 and 2.

4 Correlation Analysis and Results

The correlation analysis aims to find the relationships between cohesion and coupling metrics for object-oriented systems (OOS). The goal of this correlation analysis is to answer the following questions:

- Is there a correlation between the cohesion and coupling metrics?
- Which correlation exists between the cohesion and coupling metrics?

To answer the questions above, the non-parametric Spearman's correlation coefficient between the considered cohesion and coupling metrics was calculated.

Table 1. Descriptive statistics for the cohesion metrics

N.	Metric	Min	Max	25%	Median	Mean	75%	Std. Dev
01	LCOM1	0	7875	1	10	45	64.00	301.73
02	LCOM2	0	7875	0	3	32	46.98	278.44
03	LCOM3	0	25	1	1	2	1.64	1.77
04	LCOM4	0	25	1	1	2	1.64	1.77
05	LCOM5	0	2	1	0.82	0.97	0.77	0.47
06	CBMC	0	1	0	0	1	0.286	0.439
07	ICBMC	0	1	0	0	1	0.274	0.438
08	OL_n	0	1	0	0	1	0.286	0.439
09	PCCC	0	1	0	0.019	1	0.375	0.463
10	LSCC	0	1	0.032	0.162	1	0.372	0.407
11	CC	0	1	0.072	0.223	1	0.407	0.390
12	SCOM	0	1	0.072	0.298	1	0.450	0.403
13	TCC	0	1	0.005	0.400	1	0.472	0.409
14	LCC	0	1	0.005	0.577	1	0.535	0.426

Table 2. Descriptive statistics for the coupling metrics

N.	Metric	Min	Max	25%	Median	Mean	75%	Std. Dev
01	CBO_IUB	0	287	0	1	2	4.446	16.195
02	CBO_U	0	58	1	2	5	3.980	4.797
03	CBO	0	298	2	4	8	8.426	17.658
04	RFC	0	413	7	12	31	23.489	29.264
05	MPC	0	1739	9	22	60	53.651	102.631
06	DAC	0	131	1	2	4	3.676	6.855
07	DAC2	0	22	1	2	3	2.440	2.849
08	OCMEC	0	22	2	3	5	3.601	3.50

The result of correlation between cohesion and coupling metrics for all three systems is illustrated in Table 3 and the statistically significant (p-value $<$ 0.0001) results are underlined. They lead to the following observations (in concordance with [2,3]):

1. In CBMC, ICBMC, OL_n, PCCC, LSCC, CC, and SCOM, the coupling is always (100%) negatively correlated to cohesion. This observation is indicated by the negative signs of the results. Therefore, the results indicate that coupling and cohesion are negatively correlated.
2. Only in LCOM1 and LCOM2, the coupling is always (100%) positively correlated to cohesion. LCOM5 and coupling metrics are 87.5% positively correlated. LCOM3 and coupling metrics are 62% positively correlated and 25%

Table 3. Correlation analysis results

Metric	CBO_IUB	CBO_U	CBO	RFC	MPC	DAC	DAC2	OCMEC
LCOM1	0.33	0.36	0.44	0.52	0.53	0.43	0.44	0.60
LCOM2	0.27	0.23	0.28	0.34	0.34	0.29	0.31	0.36
LCOM3	0.07	0.06	0.08	0.07	0.06	−0.10	−0.08	0.03
LCOM4	0.07	0.06	0.08	0.06	0.05	−0.11	−0.09	0.02
LCOM5	0.13	0.10	0.11	0.13	0.13	0.08	0.07	0.02
CBMC	−0.141	−0.329	−0.309	−0.474	−0.486	−0.401	−0.407	−0.453
ICBMC	−0.141	−0.329	−0.309	−0.474	0.485	−0.401	−0.407	−0.452
OL_n	−0.141	−0.329	−0.309	−0.474	−0.486	−0.401	−0.407	−0.453
PCCC	−0.199	−0.353	−0.368	−0.552	−0.572	−0.372	−0.370	−0.445
LSCC	−0.287	−0.284	−0.330	−0.372	−0.362	−0.597	−0.599	−0.375
CC	−0.241	−0.265	−0.290	−0.334	−0.323	−0.559	−0.557	−0.345
SCOM	−0.282	−0.250	−0.298	−0.350	−0.339	−0.534	−0.544	−0.370
TCC	−0.106	−0.119	−0.134	−0.086	−0.081	−0.009	−0.016	−0.086
LCC	−0.068	−0.082	−0.083	−0.031	−0.025	0.065	0.053	−0.036

negatively correlated. While LCOM4 and coupling metrics are 50% positively correlated and 25% negatively correlated.

3. The correlations between each of LSCC, CC, and SCOM, which are the similarity-based measures, and the coupling measures are higher than the correlations between each of TCC and LCC and the coupling measures. TCC and coupling metrics are 75% negatively correlated. While LCC and coupling metrics are 37.5% negatively correlated and 12.5% positively correlated. This result is expected because the similarity based measures quantify cohesion more precisely than the metrics that ignore the degree of cohesion between a pair of methods.
4. Among all pairs of the considered cohesion and coupling metrics, the LCOM1 and OCMEC, PCCC and MPC, PCCC and RFC, LSCC and DAC1, LSCC and DAC2, CC and DAC1, CC and DAC2, have higher correlation than other considered pairs of metrics. This gives a sign that those metrics indicate cohesion or coupling more precisely and accurately than any of the other cohesion or coupling metrics considered.
5. Among all (112) relationships between the pairs of cohesion (14 metrics) and coupling metrics (8 metrics) studied on this research, 61.61% (69 pairs) were negatively correlated, 29.46% (33 pairs) were positively correlated and only 8.93% (10 pairs) were not correlated significantly. Therefore, these results indicate that cohesion and coupling in object-oriented system are negatively correlated.

The results obtained in this work are different from those obtained by Kabaili *et al.* ([24]), which stated that there was no correlation between cohesion and coupling. On the other hand, the results of the present work are similar (emphasizing) to the results obtained by Badri *et al.* ([9]), Rathore and Gupta ([32]), Al Dallal ([2]), and Al Dallal ([3]).

5 Conclusion and Future Work

In this paper, we performed an empirical study on three java open-source systems to find the correlation between cohesion and coupling metrics. In our experiments, several hundreds of classes were analyzed.

The obtained results showed that, there is in fact a negative correlation between cohesion and coupling for the studied systems. That is, if you increase the cohesion of a class (which is good), there is a greater likelihood of reducing the coupling (which is also good), and vice versa. The results also seem to indicate that the majority of classes in a system is high, the most dependence relation between cohesion and coupling is confirmed.

This research needs to expand to get a better confident generalization about the relationship between cohesion and coupling metrics, because we had some limitations to find tools for collecting several metrics. However, there are some other cohesion and coupling metrics (for example: DCO [31], Fcoh [29], ODC [20], Pcoh [21], SFC [11], CDC [19], DFC [6], EC [14], IC_OC [7,8], and RDE [30]) for which no tools were found to automate their calculation, and therefore were not included in this study.

For future work, we aim to analyze the relationship between all types of cohesion and coupling metrics. In addition, we will analyze the implications of this relation on the external quality of the system such as maintainability, reusability, and testability.

Acknowledgement. We thank Professor Jehad Al Dallal of Kuwait University, for the authorization given to make use of the tool (Quality Measuring Tool) developed by him.

Appendix

A Definitions of Cohesion Metrics:

Metric (Definition)

LCOM1, (Lack of Cohesion in Methods 1), [16].

The number of pairs of methods in the class using no instance variables in common.

LCOM2, (Lack of Cohesion in Methods 2), [17].

$$LCOM2 = \begin{cases} P - Q, if\ P - Q \geq 0 \\ 0, \quad otherwise \end{cases}$$

where, $P = number\ of\ pairs\ of\ methods\ without\ shared\ instance\ variables.$ $Q = number\ of\ pair\ of\ methods\ with\ shared\ instance\ variables.$

LCOM3, (Lack of Cohesion in Methods 3), [26].

Consider an undirected graph G, where the vertices are the methods of a class, and there is an edge between two vertices if the corresponding methods share at least one instance variable. Then $LCOM3 = |connected\ components\ of\ G|$.

LCOM4, (Lack of Cohesion in Methods 4), [23].

LCOM4 is similar to LCOM3, where graph G additionally has an edge between vertices representing methods M_i and M_j, if M_i invokes M_j or vice versa.

LCOM5, (Lack of Cohesion in Methods 5), [22].

Consider a set of methods $\{M_i\}(i = 1...m)$ accessing a set of instance variables $\{A_j\}(j = 1...a)$. Let $\mu(A_j)$ be the number of methods that reference A_j. Then $LCOM5 = \frac{\frac{1}{a} \times \sum_{1 \leq j \leq a} \times \mu(A_j) - m}{1 - m}$.

CBMC, (Cohesion Based on Member Connectivity), [15].

The CBMC for a class C, CBMC(C), is defined to be the connectivity factor of its reference graph, $F_c(G_r(C))$, scaled by the structure factor of its reference graph, $F_s(G_r(C))$. $CBMC(C) = F_c(G_r(C)) \times F_s(G_r(C)) = F_c(G_r(C)) \times \frac{1}{n} \sum_{i=1}^{n} CBMC({G_r}^i(C))$ The connectivity factor of a reference graph $G_r, F_c(G_r)$, represents the degree of the connectivity among the members, and is defined to be the ratio of the number of glue methods $|M_g(G_r)|$ to the number of normal methods $|M_n(G_r)|$: $F_c(G_r) = \frac{|M_g(G_r)|}{|M_n(G_r)|}$.

The structure factor for a reference graph $G_r, F_s(G_r)$, is defined to be the average cohesion of its children in the structure tree: $F_s(G_r) = \frac{1}{n} \times \sum_{i=1}^{n} CBMC({G_r}^i)$ where ${G_r}^i$ is one of the n children of G_r in the structure tree, and $CBMC({G_r}^i)$ denotes the cohesion of a component ${G_r}^i$.

ICBMC, (Improved Cohesion Based on Member Connectivity), [33].

The cohesion for the reference graph $G_c = (N_c, E_c)$ of a class c is defined: $ICBMC(G_c) =$

$$\left\{ \frac{|M_g|}{|N_m(G(c))| \times |N_v(G(c))|} \times \frac{1}{2} \times \sum_{i=1}^{2} ICBMC(G^i_{M_g}) \mid Q \in CS(G), \right\} \quad (1)$$

where N_m and N_v denote the set of vertexes representing methods and the set of vertexes representing instance variables of class c, respectively; G is called an elementary component of G_c. If a graph $G' = (N', V')$ is a sub graph of the reference graph G_c and its non-spanning graphs contain elementary components, then G' is called a non-elementary component of G_c. $G_c = (N_c, E_c)$ is the reference graph of a class c. Q is the minimum set of edges.

OL_n, [34].

This metric can be defined as the common strength of attributes. The strength of the attribute can be defined as the common strength of the methods that approach that attribute. Where n are the numbers of iteration which are used to calculate OL. $COM(m)$ and $COV(v)$ are used to represent the strength of method m and instance variable v respectively in Yang's cohesion measure. For a given class c, Yang first constructs a corresponding reference graph G_c. If G_c is disjoint, then let its cohesion be 0, otherwise computes its cohesion value by the following steps: (1) Initialize the iteration count variable count to 0, i.e., let count = 0; (2) For every method m_i, set its initial strength to 1, i.e., let $COM(m_i) = 1$; (3) For every instance variable v_i, let $COV(v_i) = \frac{1}{N_m(G_c)} \times \sum_{j=1}^{l} COM(m_j)$ where $N_m(G_c)$ denotes the number of method vertexes in G_c and $\lambda(m_j, c)$; (4) For every method m_i, let $COM(m_i) = \frac{1}{N_v(G_c)} \times \sum_{k=1}^{m} COM(v_k)$, where $N_v(G_c)$ denotes the number of instance variable vertices in G_c and $v_{k \in \lambda(m_j, c)}$; (5) *count* = *count* + 1, if *count* < n then Go to (3) (6) Compute the cohesion of class c, i.e., let $OL_n(c) = \frac{1}{N_v(G)} \times \sum_{r=1}^{N_v(G)} COV(v_r)$.

PCCC, (Path Connectivity Class Cohesion), [1].

$$PCCC(C) = \begin{cases} 0, & if\ l = 0\ and\ k > 1 \\ 1, & if\ l > 0\ and\ k = 1 \\ \frac{NSP(G_c)}{NSP(FG_c)}, & otherwise \end{cases}$$

Where l is the number of attributes, k is the number of methods and NSP is the number of simple paths in graph G_c; FG_c = corresponding fully connected graph.

LSCC, (Low-level design Similarity-based Class Cohesion), [4].

$$LSCC(C) = \begin{cases} 0, & if\ k = 0\ or\ l = 0 \\ 1, & if\ k = 1 \\ \frac{\sum_{i=1}^{l} x_i(x_i - 1)}{lk(k-1)}, & otherwise \end{cases}$$

Where l is the number of attributes, k is the number of methods and x_i is the number of methods that referenced.

CC, (Class Cohesion), [12].

CC = Ratio of the summation of the similarities between all pairs of methods to the total number of pairs of methods. The similarity between methods i and j are defined as: $Similarity(i, j) = \frac{|I_i \cap I_j|}{|I_i \cup I_j|}$ where I_i and I_j are the sets of attributes referenced by methods i and j, respectively.

SCOM, (Sensitive Class Cohesion Metric), [18].

SCOM = Ratio of the summation of the similarities between all pairs of methods to the total number of pairs of methods. The similarity between methods i and j is defined as: $Similarity(i,j) = \frac{|I_i \cap I_j|}{(|I_i|,|I_j|)} \times \frac{|I_i \cup I_j|}{l}$, where l is the number of attributes.

TCC, (Tight Class Cohesion), [10].

Consider a class with N public methods. Let NP be the maximum number of public method pairs: $NP = [N \times (N-1)]/2$. Let NDC be the number of direct connections between public methods. Then TCC is defined as the relative number of directly connected public methods. Then, $TCC = NDC/NP$.

LCC, (Loose Class Cohesion), [10].

Consider a class with N public methods. Let NP be the maximum number of public method pairs: $NP = [N \times (N-1)]/2$. Let NIC be the number of direct or indirect connections between public methods. Then LCC is defined as the relative number of directly or indirectly connected public methods. $LCC = NIC/NP$.

B Definitions of Cohesion Metrics:

Metric (Definition)

CBO_IUB, (CBO Is Used By), [24].

(CBO Is Used By: the part of CBO that consists of the classes using the target class): the definition of CBO merges two coupling directions: classes using the target class and classes used by the class. For changeability purposes, the former seems more relevant than the latter one, hence the split.

CBO_U, (CBO Using), [24].

(CBO Using: the part of CBO that consists of the classes used by the target class): introduced as a consequence of CBO_IUB, to cover the part of CBO not considered by CBO_IUB.

CBO, (Coupling between Object Classes), [16].

CBO for a class is the count of the number of other classes to which it is coupled. A class is coupled with another class if the method declared in one class uses method or instance variable defined by another class. CBO includes inheritance-based coupling (i.e. coupling between classes related via inheritance). For good software quality, high CBO is undesirable.

RFC, (Response for a Class), [16].

RFC can be defined as the set of methods that can potentially be executed in response to a message received by an object of that class. $RFC = |RS|$ where, *RS = the response set for the class.* The response set for the class can be defined as: $RS = \{M_i\} \bigcup_{all\,n} \{R_i\}$ where, $\{R_i\}$ is set of methods called by method M_i and $\{M_i\}$ is set of all methods in the class. *RFC of class A = Number of methods in class A + Number of distinct methods of other classes directly invoked by the methods of class A.*

MPC, (Message Passing Coupling), [27].

The MPC metric gives an indication of how many messages are passed among objects of the classes: *MPC = number of sent statements defined in a class.* The number of messages sent out from a class may indicate how dependent the implementation of the local methods is on the methods in other classes. This may not be indicative of the number of messages received by the class.

DAC, (Data Abstraction Coupling), [27].

DAC or DAC1 is the number of attributes whose types are of other classes. *DAC = number of ADTs (abstract data type) defined in a class.* The number of variables having an ADT type may indicate the number of data structures dependent on the definitions of other classes. The more ADTs a class has, the more complex the coupling of that class with other classes. The goal in designing individual software components is to represent a concept in what will eventually be an executable form. The Abstract Data Type (ADT) is the object-based paradigm's technique for capturing this conceptual information.

DAC2, (Data Abstraction Coupling 2), [27].

DAC2 is the number of distinct classes used as types of the attributes of the class.

OCMEC, (Cohesion Based on Member Connectivity), [13].

OCMEC is a number of distinct classes used as types of the parameters of the methods in the class.

References

1. Al Dallal, J.: Fault prediction and the discriminative powers of connectivity-based object-oriented class cohesion metrics. Inf. Softw. Technol. **54**(4), 396–416 (2012)
2. Al Dallal, J.: Empirical analysis of the relation between object-oriented class lack-of-cohesion and coupling. AWERProcedia Inf. Technol. Comput. Sci. **4**, 34–39 (2013)
3. Al Dallal, J.: Empirical exploration for the correlation between class object-oriented connectivity-based cohesion and coupling. Int. J. Comput. Electr. Autom. Control Inf. Eng. **9**(4), 934–937 (2015)
4. Al Dallal, J., Briand, L.C.: A precise method-method interaction-based cohesion metric for object-oriented classes. ACM Trans. Softw. Eng. Methodol. (TOSEM) **21**(2), 8 (2012)

5. Anabalon, D., Flores, A., Mateos, C., Zunino, A., Misra, S.: Controlling complexity of web services interfaces through a metrics-driven approach. In: 2017 International Conference on Computing Networking and Informatics (ICCNI), pp. 1–9. IEEE (2017)
6. Apiwattanapong, T., Orso, A., Harrold, M.J.: Efficient and precise dynamic impact analysis using execute-after sequences. In: Proceedings of the 27th International Conference on Software Engineering, pp. 432–441. ACM (2005)
7. Arisholm, E.: Dynamic coupling measures for object-oriented software. In: Proceedings of the 8th International Symposium on Software Metrics, METRICS 2002, p. 33. IEEE Computer Society, Washington, DC (2002). http://dl.acm.org/citation.cfm?id=823457.824024
8. Arisholm, E., Briand, L.C., Foyen, A.: Dynamic coupling measurement for object-oriented software. IEEE Trans. Softw. Eng. **30**(8), 491–506 (2004). https://doi.org/10.1109/TSE.2004.41
9. Badri, L., Badri, M., Gueye, A.B.: Revisiting class cohesion: an empirical investigation on several systems. J. Object Technol. **7**(6), 55–75 (2008)
10. Bieman, J.M., Kang, B.K.: Cohesion and reuse in an object-oriented system. In: ACM SIGSOFT Software Engineering Notes, vol. 20, pp. 259–262. ACM (1995)
11. Bieman, J.M., Ott, L.M.: Measuring functional cohesion. IEEE Trans. Softw. Eng. **20**(8), 644–657 (1994)
12. Bonja, C., Kidanmariam, E.: Metrics for class cohesion and similarity between methods. In: Proceedings of the 44th Annual Southeast Regional Conference, pp. 91–95. ACM (2006)
13. Briand, L., Devanbu, P., Melo, W.: An investigation into coupling measures for C++. In: Proceedings of the 19th International Conference on Software Engineering, pp. 412–421. ACM (1997)
14. Briand, L., Morasca, S., Basili, V.R.: Defining and validating high-level design metrics (1994)
15. Chae, H.S., Kwon, Y.R., Bae, D.H.: A cohesion measure for object-oriented classes. Softw.-Pract. Experience **30**(12), 1405–1432 (2000)
16. Chidamber, S.R., Kemerer, C.F.: Towards a metrics suite for object oriented design, vol. 26. ACM (1991)
17. Chidamber, S.R., Kemerer, C.F.: A metrics suite for object oriented design. IEEE Trans. Softw. Eng. **20**(6), 476–493 (1994)
18. Fernández, L., Peña, R.: A sensitive metric of class cohesion (2006)
19. Gupta, V.: Validation of dynamic coupling metrics for object-oriented software. ACM SIGSOFT Softw. Eng. Notes **36**(5), 1–3 (2011)
20. Gupta, V., Chhabra, J.K.: Object level run-time cohesion measurement. In: Information Theories & Applications, p. 136 (1993)
21. Gupta, V., Chhabra, J.K.: Package level cohesion measurement in object-oriented software. J. Braz. Comput. Soc. **18**(3), 251–266 (2012)
22. Henderson-Sellers, B.: Software metrics (1996)
23. Hitz, M., Montazeri, B.: Measuring coupling and cohesion in object-oriented systems (1995)
24. Kabaili, H., Keller, R., Lustman, F.: Class cohesion as predictor of changeability: An empirical study (2001)
25. Kumar, L., Misra, S., Rath, S.K.: An empirical analysis of the effectiveness of software metrics and fault prediction model for identifying faulty classes. Comput. Stand. Interfaces **53**, 1–32 (2017)
26. Li, W., Henry, S.: Maintenance metrics for the object oriented paradigm. In: First International Software Metrics Symposium, Proceedings, pp. 52–60. IEEE (1993)

27. Li, W., Henry, S.: Object-oriented metrics that predict maintainability. J. Syst. Softw. **23**(2), 111–122 (1993)
28. Mateos, C., Zunino, A., Misra, S., Anabalon, D., Flores, A.: Keeping web service interface complexity low using an oo metric-based early approach. In: 2016 XLII Latin American Computing Conference (CLEI), pp. 1–12. IEEE (2016)
29. Misic, V.B.: Cohesion is structural, coherence is functional: different views, different measures. In: Seventh International Software Metrics Symposium, METRICS 2001, Proceedings, pp. 135–144. IEEE (2001)
30. Mitchell, Á., Power, J.F.: An empirical investigation into the dimensions of run-time coupling in java programs. In: Proceedings of the 3rd International Symposium on Principles and Practice of Programming in Java, pp. 9–14. Trinity College Dublin (2004)
31. Morris, K.L.: Metrics for object-oriented software development environments. Ph.D. thesis, Massachusetts Institute of Technology (1989)
32. Rathore, S.S., Gupta, A.: Investigating object-oriented design metrics to predict fault-proneness of software modules. In: 2012 CSI Sixth International Conference on Software Engineering (CONSEG), pp. 1–10. IEEE (2012)
33. Xu, B., Zhou, Y.: Comments on 'a cohesion measure for object-oriented classes' by heung seok chae, yong rae kwon and doo hwan bae (Softw. Pract. Exper. **30**, 1405–1431 (2000)). Softw. Pract. Exper. 31(14), 1381–1388 (2001)
34. Yang, X.: Research on cohesion measures for classes. Ph.D. thesis, Master Thesis, Department of Computer Science & Engineering in Southeast University (2002)

Data Analysis Algorithm for Click Fraud Recognition

Marcin Gabryel[1,2(✉)]

[1] Institute of Computational Intelligence,
Czestochowa University of Technology,
Al. Armii Krajowej 36, 42-200 Częstochowa, Poland
marcin.gabryel@iisi.pcz.pl

[2] Spark DigitUp Sp. z. o. o., Plac Wolnica 13 lok. 10, 31-060 Kraków, Poland

Abstract. This paper presents an analytical system designed to detect click fraud on the Internet. The algorithm works with the data collected from an advertiser's website to which the Pay-Per-Click traffic is directed. This traffic is not entirely carried out by humans, as a large part of it is carried out by bots – software running automated tasks. The purpose of the proposed algorithm is to analyze the data of individual clicks coming from advertisements and to automatically classify them as suspicious or correct. The paper presents the mechanisms of comparing different types of data, their classification and the tuning of particular elements of the algorithm. Results of the experimental research confirming the effectiveness of the proposed methods are also presented.

Keywords: Click fraud · Classification · Ad fraud detection

1 Introduction

Pay-Per-Click (PPC or Cost Per Click, CPC) is by far the most popular model used in online advertising. It is used by giants such as Google, Facebook, Bing and others. Despite its great popularity and security features, this model can be still abused in a number of ways. These frauds involve fabricating large numbers of clicks on online adverts. This is done either directly by organic users or by non-organic users, for example special bot software. Fake clicks generate additional costs to be charged to advertisers. It is estimated that only in 2017 every fifth click was fraudulent [3], and their number is increasing with every month. Such unfair practices are most often carried out by competitors. Quickly "clicking" on an advertisement makes it disappear and a fraudulent competitor has a better chance of showing their advertisement in subsequent editions instead of the targeted victim's advertisement. Publishers can also make additional clicks in order to earn more money in a shorter time from the advertisement on their website.

This type of fraudulent activity falsifies the analytical data on which the advertiser's marketing decisions are based. When an advert is displayed and subjected to fake clicks, analytical data clearly indicates that the click-through rate is very high and the conversion rate is very low. Since there is a chargeable fee per click, the cost of advertising will be higher than the profit. In such a case, such form of advertising

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 437–446, 2018.
https://doi.org/10.1007/978-3-319-99972-2_36

should be abandoned. However, if it is possible to obtain additional data which is being monitored on an ongoing basis, then it is quite likely that a changeable behaviour of the clicks on an advertisement will be noticed. One of the features that can be observed is a certain repeatability in the data of a customer from whom there has been multiple connection with the advertiser's website.

This article presents possibilities of monitoring and analyzing data after clicking on a given advertisement and switching to its advertiser's target website. As a result of the operation of the presented algorithm, information about a given click is obtained - the click is either confirmed as valid or treated as suspicious. The statistics collected in such a way about all clicks allow to settle accounts with the company organizing the advertising campaign and to make an appropriate complaint. An example is Google, which did recognize the problem of click fraud despite application of security features. Hence, they have decided to return some of the payments to advertisers for the losses caused by bots aiming at generating fake traffic [5].

The analytical activities described are based on an appropriate method of monitoring and obtaining relevant data enabling identification of the user or bot. For this purpose, a special JavaScript is inserted on the advertiser's website, which collects information about the equipment, browser used, user behavior, connection parameters and detects irregularities related to the operation of the browser and the JavaScript language already at the very time of creating the website by the browser. These data are sent to a server, where they are analyzed later.

During the development of an appropriate analytical algorithm, it proved very difficult to develop a method for comparing the data obtained during the monitoring of users as the data comes in different types - text, number, index and others. For this purpose, an algorithm known for text search, TF-IDF (Term Frequency – Inverse Document Frequency), was chosen. The TF-IDF algorithm is used to create an index and to quickly search for text documents. The algorithm is also used as a method to assess the relevance of a document, for example, in search engines. In the case being discussed, terms will be the values of particular features obtained from the website after clicking, and the information about a single entry to the website will be treated as a document. Analyzing the main assumptions of the TF-IDF algorithm, it can be noticed that the greater the number of similar terms in documents, the less important they are for the final comparison of two documents. In this case, there will always be one term in a single document in a given feature. This term corresponds to the value of this feature (e.g. the user's IP number, the value of fingerprint or the number of pages visited previously in the browser's history). However, it is possible to artificially increase the number of occurrences of such term in a single document. The number of occurrences of the term can be used to assign weights to individual features. If we assume that a given value of a feature is more frequent (i.e. its weight is greater), it will be treated as equivalent to the fact that the TF-IDF algorithm recognizes this feature as being of a lesser significance. Conversely, a smaller value for the weight of a feature will give it a greater significance. In the proposed approach, the choice of these weights is made by the Differential Evolution (DE) algorithm.

The value of the comparison obtained with the use of the TF-IDF algorithm allows us to use e.g. clustering or classification algorithms, where one of the operations is to make comparisons between individual input data. For the presented analysis, a simple but effective *k*-NN algorithm (*k*-nearest neighbors algorithm) has been proposed. The operation of this algorithm depends on finding *k* closest (most similar) data to the tested sample. The sample tested is given the label which is the most numerous of the labels belonging to the nearest *k* data.

The issue of click fraud is discussed rather rarely in the literature. The study analyzing the operation of non-organic user clicking on advertisements on the Internet is available in [9]. It studied the behavior of bots for four different advertising systems offered by Google, Facebook, LinkedIn and Yahoo. The book [10] comprises most of the available information about abuse found in online advertising, including the mechanisms of fraud in the PPC model. A lot of work, e.g. [11, 12] uses server logs obtained directly from web servers and analyzes them in search of bot farms or botnets. In [15] the FCFraud system is presented. FCFraud tries to detect programs that execute in the background, implement browser functionalities and perform click-fraud stealthily. Paper [16] studies interesting problem of detecting frauds in comparison-shopping services – patterns and anomalies in user click behaviors. However, it still remains difficult to find real algorithms that can help in organic versus non-organic user classification.

The paper has been divided into several sections. After the introduction and brief description of the algorithms used (Sect. 2), Sect. 3 describes in detail the algorithm, the course of the analysis and optimization of the process. The experimental studies showing the effectiveness of the presented method are discussed in Sect. 4. The last Section contains conclusion and suggestions for further work on the algorithm.

2 Algorithms Used in the Proposed Solution

2.1 TF-IDF

The TF-IDF [1] algorithm is a dedicated algorithm for quick document searching. In the proposed algorithm, it is used to calculate the measure of similarity between the data describing accessing a given website. The TD-IDF algorithm is designed to calculate the frequency of occurrence of specific words in a given document, taking into account its occurrence in all documents. This value is computed from the following formula:

$$(\text{tf-idf})_{i,j} = \text{tf}_{i,j} \times \text{idf}_i \tag{1}$$

On the left side of the equation you can see two components i.e. tf and idf. The first is the frequency of occurrence of the word expressed by the formula:

$$\text{tf}_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{2}$$

where $n_{i,j}$ is the number of occurrences of term t_i in document d_j. The denominator, on the other hand, contains the sum of the number of occurrences of all the words in the selected document i.e. document d_j. The second value - idf_i - from Eq. (1) is expressed by the following formula:

$$\mathrm{idf}_i = \log \frac{|D|}{|\{d : t_i \in d\}|} \tag{3}$$

where $|D|$ it is the total number of documents, and $|\{d : t_i \in d\}|$ is the number of documents containing at least one occurrence [1, 2].

2.2 Differential Evolution

Differential Evolution is an optimization algorithm that belongs to the group of evolutionary algorithms [6]. The DE algorithm is very simple; however, it works surprisingly effectively. DE operates on population $\mathbf{X}^G$ of individuals $\mathbf{X}_i^G$, where $i = 1, \ldots, NP$, NP – population size, and $\mathbf{x}_i^G = \left[x_1^G, x_2^G, \ldots, x_D^G\right]$, G is the number of the generation. In the algorithm, all the individuals are D – dimensional vectors of the real numbers, where the phenotype is identical to the genotype. Mutation in this method is based on random permutation of each vector $\mathbf{x}_i^G$ by differencing two other vectors from population $\mathbf{X}^G$ multiplying them by a constant value. For each vector $\mathbf{x}_i^G$ mutant vector $\mathbf{v}_i^{G+1}$ is generated according to:

$$\mathbf{v}_i^{G+1} = \mathbf{x}_{r1}^G + F \cdot \left(\mathbf{x}_{r2}^G - \mathbf{x}_{r3}^G\right) \tag{4}$$

with random indexes $r1, r2, r3 \in \{1, 2, \ldots, NP\}$, $r1 \neq r2 \neq r3 \neq i$ and $F \in [0, 2]$ is a parameter to control the mutation and it is real and constant factor.

A crossover operator is introduced to mix random elements of parent vector $\mathbf{x}_i^G$ and elements of the vector $\mathbf{v}_i^{G+1}$ after mutation, whereas the output receives a trial vector:

$$u_{ji}^{G+1} = \begin{cases} v_{ji}^{G+1} \text{ if } rnd_j \leq CR \text{ or } j = d_i \\ x_{ji}^{G+1} \text{ if } rnd_j > CR \text{ or } j \neq d_i \end{cases}, j = 1, \ldots, D \tag{5}$$

where $CR \in [0, 1]$ is the crossover constant, rnd_j is a uniform random number generator with outcome $\in [0, 1]$ and $d_i \in 1, 2, \ldots, D$ is a randomly chosen index. The crossover operation refers to optimization process. Update the value of $\mathbf{x}_i^{G+1}$ in the next generation:

$$\mathbf{x}_i^{G+1} = \begin{cases} \mathbf{u}_i^{G+1} & \text{if } f\left(\mathbf{u}_i^{G+1}\right) < f\left(\mathbf{x}_i^G\right) \\ \mathbf{x}_i^G & \text{if } f\left(\mathbf{u}_i^{G+1}\right) \geq f\left(\mathbf{x}_i^G\right) \end{cases} \tag{6}$$

where $f(\cdot)$ is a fitness function. The description in pseudo-code of applied DE is presented below:

1. Initiate the algorithm:
   a. Define all coefficients: NP, F, CR.
   b. Define fitness function $f(\cdot)$
   c. Create random population $\mathbf{x}_i^G$ in the solution space.
   d. Set the generation number $G = 0$.
2. Repeat steps 3 and 4 until $G \leq generation$.
3. For each vector $\mathbf{x}_i^G$ from population $\mathbf{X}^G$:
   a. Generate mutant vector $\mathbf{v}_i^{G+1}$ according to (4).
   b. Crossover vectors within the population according to (5).
   c. Change the values $\mathbf{x}_i^{G+1}$ according to (6).
4. Next generation $G = G + 1$.

The values from the last population with the best fitness are an optimal solution.

2.3 *k*-Nearest Neighbors

The *k*-Nearest Neighbors (*k*-NN) method belongs to the group of lazy algorithms, i.e. those that do not create an internal representation of the teaching data, but seek a solution only when a testing standard appears. This algorithm stores all the teaching patterns for which it determines the distance of the test pattern. The classification process consists in determining k neighbors to whom the tested element (asterisk) is closest to the selected metric, and then determining the result based on the majority vote.

3 Proposed Algorithm

The presented algorithm has been prepared for the purpose of analyzing data collected during the process of making clicks on advertisements and price comparison engines. The result of the algorithm is information about whether the behavior on the website is natural, generated by man or generated artificially by a computer program (bot). A data flow diagram is presented in Fig. 1. By clicking on an advertisement on a publisher's website, the user is taken to the advertiser's website. When the website opens in the client's browser, the algorithm instantaneously starts a script monitoring the client's behavior. The downloaded data collected in this process are then sent to a database where it is analyzed and the algorithm that performs this analysis is the one presented in this paper. The result of the analysis is sent to the advertiser.

The data is collected by the JavaScript installed on the advertiser's website. All publicly available information that can be downloaded from the browser is collected. The most important downloaded data include the following:

– the class of the IP address,
– the internet service provider,

– the fingerprint - an identifier based on various parameters available from the JavaScript language in the browser [4],
– the features of the visit:
  • the number of pages visited,
  • number of pages in the browsing history,
  • number of visits,
  • the header of the http User-agent,
  • the heading http Referrer,
– the parameters of the browser or of the computer equipment:
  • screen resolution,
  • screen orientation (vertical, horizontal),
  • information on the language,
– traps to detect anomalies in the browser or in the equipment,
– user behavior:
  • mouse movements monitoring data,
  • monitoring page scrolling,
  • keystroke data.

The data are collected from each of the websites visited by the user and are then linked to each other in order to obtain a complete picture of the behavior of a human or a bot (organic or non-organic user) when visiting the advertiser's website during one session.

The algorithm performing the analysis can be presented in several steps:

1. Preparation of the TF-IDF matrix on the basis of the data collected so far. In addition, any existing data are marked by the expert as correct or suspicious.
2. The algorithm is adapted by weighting each characteristic feature. The DE algorithm is used for optimization. The result of the classification of the k-NN algorithm and the comparison with the expert opinion shall be considered as a fitness function.
3. The tuned analysis algorithm is run for each new sample that enters the system.

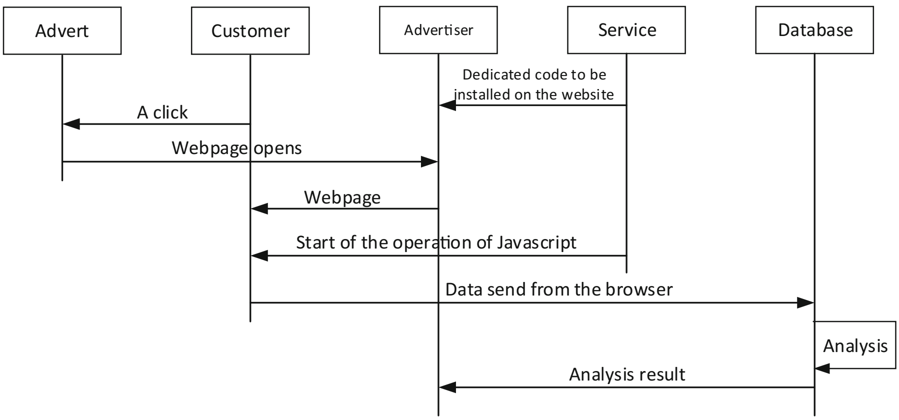


Fig. 1. Data flow diagram for fraud click analysis.

Each of the parameters sent from the customer is treated as separate feature c_k, where $k = 1, \ldots K$, K – the number of all features. Parameters make up the description of whole document d_j. Each feature has a value – term t_{i_m,c_k}, where i_m – are m-th term belonging to feature c_k, $m \epsilon [1, \ldots, T_{c_k}]$, T_{c_k} – the number of terms in feature c_k. Hence, the document is vector K of terms $d_j = \left[t_{i_1,c_1}, t_{i_2,c_2}, \ldots, t_{i_K,c_K}\right]$. The TF-IDF algorithm needs to be a little adjusted for the sake of the features occurring in the document. Coefficient $tf_{i_m,c_k,j}$ for each term i_m belonging to feature c_k in document j is calculated in the following way:

$$tf_{i_m,c_k,j} = \frac{n_{i_m,c_k,j}}{\sum_{v,w} n_{i_v,c_w,j}} \tag{7}$$

where $n_{i_m,c_k,j}$ is the number of occurrences of term t_{i_m,c_k} in document d_j. Then, idf needs to be calculated according to the formula:

$$\mathrm{idf}_{i_m,c_k} = \log \frac{|D|}{\left|\left\{d : t_{i_m,c_k} \in d\right\}\right|} \tag{8}$$

Since $n_{i_m,c_k,j}$ takes the value of 1, for each of the features c_k it is possible to use weight, which will determine the significance of each of the weights in the final calculations of TF-IDF algorithm. Formula (7) can thus be modified as follows:

$$tf_{i_m,c_k,j} = \frac{n_{i_m,c_k,j} \cdot w_k}{\sum_{v,w}\left(n_{i_v,c_w,j} \cdot w_k\right)}, \tag{9}$$

where w_k is a weight of k-th feature. The comparison of the two documents is made by a cosine similarity measure according to the following formula:

$$sim\left(d_{j_1}, d_{j_2}\right) = \frac{\sum_{k=1}^{K} tf_{i_k,c_k,j_1} tf_{i_k,c_k,j_2}}{\sqrt{\sum_{k=1}^{K} tf_{i_k,c_k,j_1}} \sqrt{\sum_{k=1}^{K} tf_{i_k,c_k,j_2}}} \tag{10}$$

The selection of weights w_k is made by applying the DE algorithm (Sect. 2.2). The fitness function is based on the accuracy rating [8]. Accuracy is the ratio between data that are correctly classified and the total number of samples, as calculated using the equation:

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \tag{11}$$

where: TP, TN, FN and FP are True Positive, True Negative, False Negative and False Positive, respectively. After weight optimization, the algorithm can work automatically and rate subsequent links to the advertiser's website made by clicking.

4 Experimental Results

As test data, authentic data were collected from the website of a large online shop operating in Poland (approx. 30 thousand data). In the PPC model, the online shop settles the clicks coming from a price comparison engine. The aim of the research was to analyze clicks and test the work of the algorithm to automatically make decisions about who the click comes from - from an organic or non-organic user.

The data is divided into 3 parts - a teaching part (80% of all samples), a validation part (10% of the total), used during the operation of an evolutionary algorithm to calculate the accuracy value as a fitness function, and a test part (10%) for the final verification of the whole algorithm. Each of the data was labelled ('ok', 'suspect') during a manual evaluation by an expert. The results obtained during the following algorithm operation points are presented in Table 1 as percentage values. It is clear that assigning a weight to each individual attribute gives much better classification results than using the pure TF-IDF algorithm alone.

Table 1. Results obtained during the operation of the analytical algorithm before and after optimization.

	Training data	Validating data	Testing data
Before optimization	82.71	82.71	82.71
After optimization	**97.09**	**97.11**	**97.19**

The next step is to select the appropriate value of k in the k-NN algorithm. Subsequent tests for the weights obtained earlier are presented in Table 2. The following rows show the percentage accuracy values obtained for various values k. The diagram depicting the accuracy efficiency for different data type (training, validating and testing) in relation to values k. are presented in Fig. 2.

Table 2. Change of the accuracy parameter value depending on the k-NN algorithm value.

Value k	Training data	Validating data	Testing data
5	97.87	97.68	97.71
10	97.09	97.11	97.19
15	97.27	97.16	97.16
20	96.83	96.75	96.73
25	96.85	96.75	96.73
30	96.57	96.39	96.33

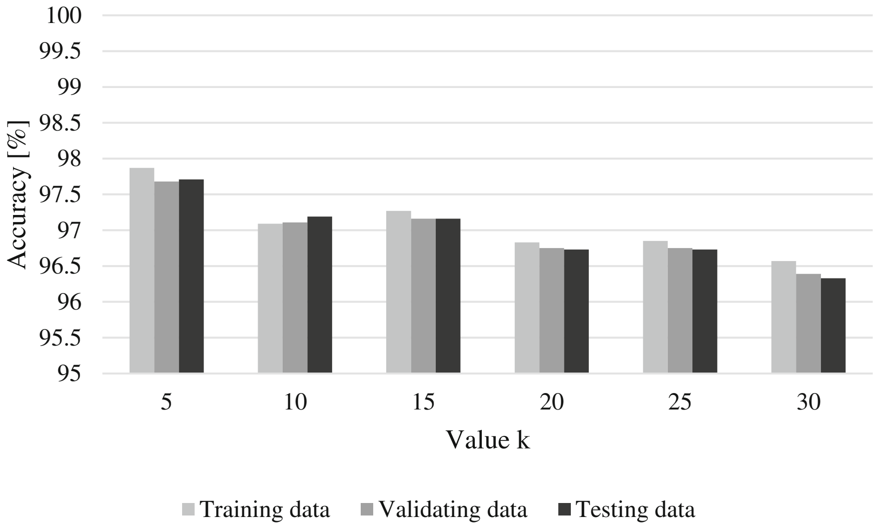


Fig. 2. Accuracy classification efficiency [%] for different data type in relation to value *k*.

5 Conclusions

The article presents an analytical algorithm designed to evaluate a click on the advertiser's website as a click made by a man or a bot (an organic or non-organic user). A special JavaScript was installed on the advertiser's website which collected data from various customers for a month. The operator/expert can evaluate subsequent clicks manually. However, it requires tedious work and does not ensure on-line action in response to the traffic on the advertiser's website.

The data that is downloaded from the website when displayed and used are of different types. In order to be able to compare them with each other, it is necessary to develop an algorithm and use an appropriate method of comparison. The article presents a modified TF-IDF algorithm, whose work is adjusted by an evolutionary algorithm. The results, combined with the application of the *k*-NN classifier, turned out to be very good. In the future, the presented algorithm can be extended by other elements, such as revealing time relationships between clicks, regularity in behavior which is unusual for human users or other classifier described in [13, 14, 17].

References

1. Salton, G., Buckley, C.: Term-weighing approaches sin automatic text retrieval. Inf. Process. Manage. **24**(5), 513–523 (1988)
2. Ramos, J.J.: Using TF-IDF to determine word relevance in document queries. Technical report, Department of Computer Science. Rutgers University (2000)
3. http://blog.pixalate.com/desktop-ad-click-fraud-rising-stats-data-2017. Accessed 13 June 2018

4. https://github.com/Valve/fingerprintjs2. Accessed 13 June 2018
5. https://support.google.com/adwords/answer/42995?hl=en. Accessed 13 June 2018
6. Storn, R., Price, K.: Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. J. Global Optim. **11**(4), 341–359 (1997)
7. Cover, T., Hart, P.: Nearest neighbor pattern classification. IEEE Trans. Inf. Theory **13**(1), 21–27 (1967)
8. Olson, D.L., Delen, D.: Advanced Data Mining Techniques, 1st edn. Springer, Heidelberg (2008)
9. Neal, A., Kouwenhoven, S., Sa, O.: Quantifying online advertising fraud: Ad-click bots vs humans. Technical report, Oxford Bio Chronometrics (2015)
10. Zhu, X., et al.: Fraud Prevention in Online Digital Advertising. Springer, Cham (2017)
11. Seyyar, M.B., Çatak, F.Ö., Gül, E.: Detection of attack-targeted scans from the Apache HTTP Server access logs. Appl. Comput. Inf. **14**(1) (2018)
12. AsSadhan, B., Moura, J., Lapsley, D., Jones, C., Strayer, W.: Detecting botnets using command and control traffic. In: Eighth IEEE International Symposium on Network Computing and Applications, NCA 2009, pp. 156–162 (2009)
13. Woźniak, M., Połap, D.: Bio-inspired methods modeled for respiratory disease detection from medical images. Swarm Evol. Comput. **41**, 69–96 (2018). https://doi.org/10.1016/j.swevo.2018.01.008
14. Woźniak, M., Połap, D.: Adaptive neuro-heuristic hybrid model for fruit peel defects detection. Neural Netw. **98**, 16–33 (2018). https://doi.org/10.1016/j.neunet.2017.10.009
15. Iqbal, M.S., Zulkernine, M., Jaafar, F., Gu, Y.: Protecting internet users from becoming victimized attackers of click-fraud. J. Softw. Evol. Process **30**(3), e1871 (2018)
16. Lee, S.C., Faloutsos, C., Chae, D.K., Kim, S.W.: Fraud detection in comparison-shopping services: patterns and anomalies in user click behaviors. IEICE Trans. Inf. Syst. **100**(10), 2659–2663 (2017)
17. Damasevicius, R., Valys, R., Wozniak, M.: Intelligent tagging of online texts using fuzzy logic. In: 2016 IEEE Symposium Series on Computational Intelligence, SSCI 2016 (2017). https://doi.org/10.1109/ssci.2016.7849917

Information Technology Applications: Special Session on Smart e-Learning Technologies and Applications

Competence Management in Teacher Assignment Planning

Eryk Szwarc[1(✉)], Irena Bach-Dąbrowska[2], and Grzegorz Bocewicz[1]

[1] Faculty of Electronics and Computer Science, Koszalin University of Technology, Sniadeckich 2, 75-453 Koszalin, Poland
eryk.szwarc@tu.koszalin.pl

[2] Finance and Management Faculty, WSB University of Gdansk, al. Grunwaldzka 238A, 80-266 Gdansk, Poland
ibach@wsb.gda.pl

Abstract. In selecting teachers for courses, effort is always made to ensure sound use of competences to achieve the desired instructional quality under the assumed cost conditions. The present study is a review of the state of the art in research on competence management problems, in particular, the Teacher Assignment Problem (TAP). The article focuses methods of modelling TAP, competence models, and level of competence as a function of time.

Keywords: Teacher assignment problem · Competence
Competence management

1 Introduction

Everyday practice and the available literature of the subject [1–3] show that the organization and planning of teaching that takes place every year in higher education institutions is associated with the problems of assignment of teachers to courses and scheduling/timetabling of those courses. Usually, teachers are assigned to courses before scheduling [2–7]. A few studies [8], however, suggest that a reverse approach can also be applied. In this case, the main difficulty lies in matching teachers' preferences regarding courses, contact hours, etc., and their teaching competences to the existing schedule. There are also numerous approaches in which teacher assignment and scheduling are done simultaneously [8–12]. The schematic in Fig. 1 illustrates the ideas behind each of the three approaches.

This paper focuses on the problems of teacher assignment associated with stage 1 of approach A (Fig. 1). One of the key criteria for teacher assignment are teachers' competences. Competences are understood as a set of elements that allow an instructor to teach classes within a specific field of instruction; they include knowledge, experience, and skills. The individual competences of individual teachers make up the competence structure of a university's academic staff. The staff's competences naturally determine what type of courses can be taught at the given institution and the costs of running those courses.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 449–460, 2018.
https://doi.org/10.1007/978-3-319-99972-2_37

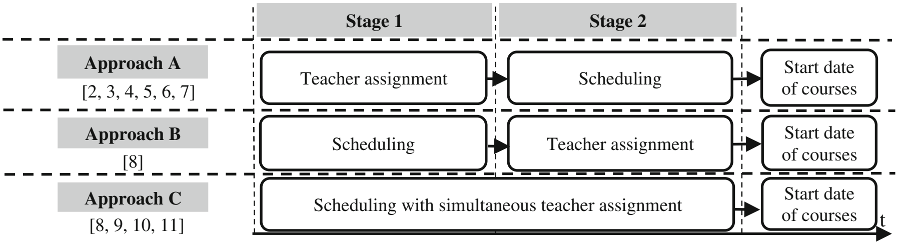


Fig. 1. Different approaches to the organization of academic courses

In this context, models which take account of competence structures of academic staff allow to answer typical questions in the area of analysis of competence structures related to the assessment of the human resource potential of a given organizational unit (institute/faculty/university) and estimation of the costs of teacher assignment, e.g.:

> Does a given set of teachers who have specific competences ensure assignment of teachers to courses at the same time satisfying the assumed constraints? If so, what is the cost of such an assignment?

These models also allow to answer questions in the area of synthesis of competence structures related to the possibility of strengthening an institution's human resource potential (the literature offers only a few publications in this topic area), e.g.,

> Will teacher refresher training ensure a teacher assignment that will satisfy the assumed constraints? If so, which teachers should be trained in what competences?

Acquisition of competences takes time and can generate costs. This means that questions related to the enhancement of the teaching potential of the available staff can take the following form:

> How long will the refresher training last and/or how much will it cost? Which employees should be additionally trained to achieve the desired teacher assignment in the shortest possible time and/or at the lowest possible cost?

It is assumed that each employee acquires competences at a different rate and with a different effectiveness. This is associated with the fact that each person has a different temperament and personality, etc. Besides, the pace of learning is influenced by what competences a person already has. In other words, employees with competencies in a given area acquire new competences in this area easier/faster than employees who do not have any competences in this field. This phenomenon has so far been investigated in research on the acquisition of competences by employees in the area of production [13] and in project-oriented organizations [14]. Discussions of this topic are necessary to fill in the substantial gap in teacher assignment research.

From this angle, the purpose of the present review of selected models and methods used in solving teacher assignment problems is to analyze the ways in which the development of professional competences of academic staff can be modelled and planned.

The next section presents a formulation of the teacher assignment problem. The main part of the article is Sect. 3, which presents the state of the art as documented in the literature of the subject. The last part of the review (Sect. 4) contains conclusions and suggestions for future research.

2 Formulation of the Teacher Assignment Problem

In the general case, the teacher assignment problem boils down to allocating resources (teachers) to activities (classes). A well-prepared assignment must guarantee the satisfaction of constraints related to the specific contact hour limits, minimum academic staff complement, courses taught by instructors with an appropriate academic degree or title, etc. In further discussion, the following formal definition of the **Teacher Assignment Problem (TAP)** is adopted, which accents the role of the competence structure of the available teaching staff.

Given is a Set of Courses to be Taught in a Given Academic Year. The courses are run in the winter or the summer semester. There are various course formats (lectures, tutorials, labs, workshops, seminars). Each course (taught in a specific format) has a fixed number of course sections or classes. Students in each course section have to attend a fixed number of course meetings (colloquially called classes). Each course meeting has an assigned number of teaching credits.

Given is a Set of Teachers. Each teacher (instructor) has a set of competences (skills, qualifications) to teach specific courses/course meetings. The set is described in a binary way (can/cannot teach a specific course). The competences of individual teachers make up the **competence structure** of the entire staff (set of teachers).

The staff members' competences can change. The resulting **change in the structure** is understood as the acquisition by at least one staff member of new competences allowing them to teach a specific course (group of courses).

In addition, each teacher has a professional degree/title (professor, doctor, master, engineer) and is assigned a specific number of teaching credits (teaching load, credit hours) per academic year.

A **permissible teacher assignment** is understood as an assignment of teachers to course meetings which satisfies the following **constraints:**

- each course meeting can be conducted by only one competent teacher,
- the set of all course meetings for a specific course section (group) can be run by no more than three teachers,
- lectures and seminars can only be delivered by professors and doctors,
- each teacher should have a guaranteed teaching quota,
- all course meetings must be assigned to teachers,
- others (following from the individual needs of a university/organizational unit).

This list, of course, does not include all possible constraints, such as constraints on daily/weekly/monthly working hours (which are relevant to the scheduling problem, stage 2, approach A from Fig. 1).

Given this model, the following **questions** can be formulated:

1. Does there exist a permissible teacher assignment?
2. Does there exist a competence structure that guarantees a permissible assignment of teachers to courses?
3. Does the given number of changes to the competence structure guarantee a permissible assignment of teachers to courses?
4. What minimum number of changes to the competence structure guarantees a permissible teacher assignment?

A **TAP** formulated for these assumptions may have a decision-making character (questions 1–3) or an optimization character (question 4). It should be noted that questions 2–4 are associated with the synthesis of the competence structure.

In the next section, a comparison is made among the existing models of the problem of assigning teachers to courses. Similarities and differences between them and the TAP model are shown. Attention is focused on ways of modelling the competences of academic staff.

3 State of the Art

The TAP defined in the previous chapter is a problem that relates to stage one of approach A (Fig. 1). Different variants of this problem are subject of intensive research both in the area of management [15, 16] and IT [2, 4, 5, 8, 10, 17]. TAP [2, 5, 6, 8, 18] is also, less often, referred to as the Teaching Service Distribution (TSD) problem [4]. There are also studies in which teacher assignment is treated as a synonym of timetabling or scheduling. Moreover, there are many publications which classify the problems of teacher assignment as scheduling problems. In this light, it seems necessary to make a systematic review of concepts and locate the TAP in the above-mentioned classifications.

One example of taxonomy of teacher assignment problems has been proposed in paper [20], which is a review of approaches to the University Course Timetabling Problem (UCTTP). The general UCTTP has been defined there as the problem of allocating course meetings (taking into account data regarding teachers, students, and courses) to time windows (day and/or week) and rooms (taking into consideration the required equipment, seating capacity, etc.), assuming that teacher assignment to courses is already known (the planner who makes the timetable works with a ready teacher assignment). This means that TAP is solved before UCTTP (Fig. 2).

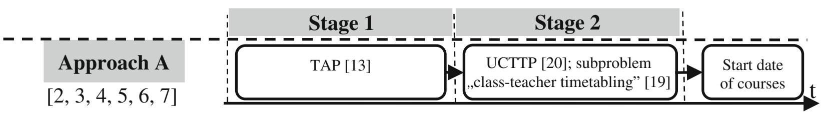


Fig. 2. TAP and UCTTP on the academic timetabling courses approaches shown in Fig. 1

A classification of UCTTP is shown in Fig. 3a. It is easy to notice that TAP is not included in it. However, to the classification given in [19], where TAP figures in the class of scheduling problems, it is immediately clear that it is possible to extend the classification presented in Fig. 3a to take into account the classification from Fig. 2 (see Fig. 3b). In other words, TAP can be included in the class of university course scheduling problems, but unlike the problems encountered in the literature, it offers the possibility of both analysis and synthesis of competence structures of the teaching staff. These possibilities are described more broadly in Subsects. 3.1 and 3.2.

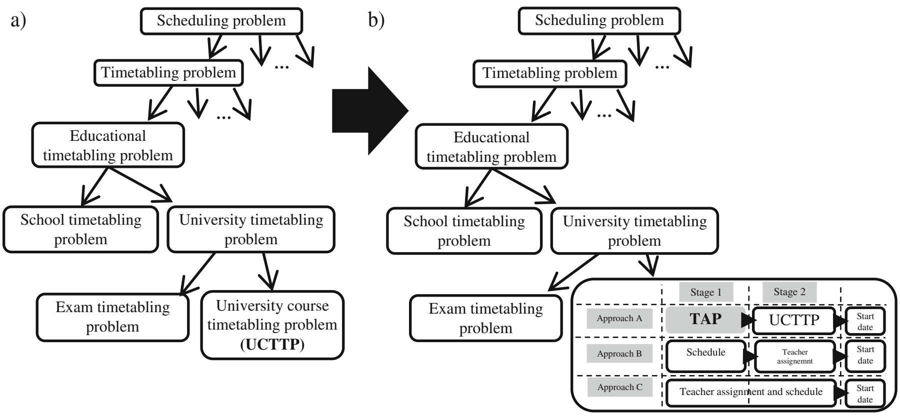


Fig. 3. An extended UCTTP classification diagram

3.1 Models of the Teacher Assignment Problem

TAP emphasizes the role of the structure of competences of the available teaching staff, i.e. the role these competences play as constraints on selecting teachers for particular courses. As the literature of the subject shows, there are TAP/TSD models which do not take into account the competences of teachers [2, 3, 6, 10, 21], as well as those which treat competences as the key condition for assigning instructors to courses [4, 5, 8, 18]. It should be noted that the most frequently chosen modelling method uses just binary competence structures, in which the instructor is assumed to either have the necessary competences to teach particular courses (logical value “1”) or not (logical value “0”). The approach under discussion is also seen in models in which competences of teachers are described in the following way: a given teacher can/cannot teach a specific group of courses. So modelled, competences determine an instructor’s eligibility for teaching a given course.

Some models [2–4, 8, 10, 18, 22–24], take into account teachers’ preferences for specific courses (e.g., John Smith wants to teach Mathematical Analysis, but he does not want to run a course in Functions of a Complex Variable) and the number of hours he/she wants to teach (e.g. Smith wants to teach not more than 300 h in the academic year). It is intuitively obvious that the fulfilment of a teacher’s preferences affects the level of their satisfaction. Hence, most research is focused on models and methods that allow to maximize teacher preferences [2–4, 8, 10, 18, 23, 24]. In this context, it is worth noting that models of TAP/TSD are, generally, dedicated to many different goals, such as minimizing the number of courses assigned to an instructor [6], assigning a balanced number of contact hours to each teacher [5, 8, 18, 21], etc.

In general, TAP/TSD models incorporate data on:

- course structure:
  - course type/form: lecture, seminar, problem-solving class, tutorial, lab session,
  - number of course sections,
  - number of contact hours for a course section,
  - non-teaching projects (counted as additional contact hours),
- teachers:
  - academic degree/academic title: professor, assistant professor, doctor, junior lecturer, trainee, etc.,
  - a teacher's "importance index" based on the position he/she holds,
  - teaching credits – number of hours a teacher has to teach in a year; number of course sections to be allocated,
  - historical data: what teacher taught what courses in what period of time (semesters),
- teacher competence and preference structures:
  - binary defined competences/qualifications required to run a particular course (the teacher has or not has appropriate qualifications/competences to teach a given course and course type),
  - scientific areas, binary assigned to teachers, associated with a given group of courses (the teacher is qualified to run a group of courses)
  - preferred total number of contact hours in an academic year,
  - preferred number of contact hours to be spent teaching a particular course type,
  - a course preference list.

The constraints used in the models take into account the various requirements following from the specific character of a given university or the legislative norms in force in a given country. That is why some of the constraints are common to all models, and some are optional. The former group of constraints includes, among others, the following:

- each course must be assigned to at least one teacher,
- each teacher should have a guaranteed teaching quota,

The latter group encompasses the following constraints:

- courses can only be taught by competent instructors,
- a given course type can only be taught by a staff member with a specific academic degree (e.g. a seminar can only be taught by a professor, a lecture can be delivered by a professor and an assistant professor, etc.),
- given is the maximum number of teachers who can teach a given course section (in practice, one course section is often taught by several teachers),
- a teacher cannot teach a course s/he had taught previously for X semesters,

- a teacher assignment should guarantee a balanced workload for employees (in planning courses, planners often take care to assign a similar number of contact hours to each staff member),
- to reduce the time needed to prepare for classes, the number of courses assigned to individual teachers is limited.

The questions that are most often formulated when these data and constraints are adopted are the following:

- What teacher assignment will maximally meet teachers' preferences? – see: [2, 4, 8, 10, 18, 22–24]
- What teacher assignment will minimize deviations from teachers' preferred number of contact hours? – see: [4]
- What teacher assignment will minimize the difference between the teachers with the smallest and the largest workloads? – see: [5, 10, 21, 23, 24]
- What teacher assignment will minimize the number of different courses assigned to one teacher? – see: [5, 6]

It is easy to notice that to answer the above questions it is necessary to analyze the adopted competence structure, i.e. to assess what effect (teacher assignment plan) can be achieved given a set of teachers with a specific competence structure. A different approach is related to questions 2–4 considered in Sect. 2, which boil down to the question: what competence structure will allow to achieve the expected effect?

In this context, research reported in [25–27] looks at the issues of teacher assignment from the perspective of synthesis of competence structures, as an alternative to the analysis of those structures. The model of TAP presented in those studies allows one to search for competence structures (changes to such structures) which will guarantee a teacher assignment satisfying the adopted constraints. This search consists in looking for answers to the following questions:

- Does the given teaching staff have a competence structure that guarantees permissible assignment of teachers to courses?
- Does the given number of changes to the competence structure guarantee permissible assignment of teachers to courses?
- What minimum number of changes to the teaching staff's competence structure guarantee a permissible teacher assignment?
- Which teacher should be trained in what competences to guarantee a permissible teacher assignment?
- etc.

The answers to the above questions do not indicate what time is needed to change the competences of the individual teachers. However, if one knows the time required to make the necessary changes (training, retraining, etc.), one can formulate a new category of questions: How long will the training (change of the competence structure) of the specific employees take? Which employees should be trained in what competences for an institution to be able to assign teachers to courses in the shortest possible period of time? When should changes in the structure of the staff's competences be started to ensure that there are enough instructors to teach courses that begin at a given time? The next section is an overview of existing competence models.

3.2 Competence Models

In the literature of the subject, competences are defined in various ways. In [28], competences are understood as the general capability based on knowledge, experience, values, dispositions which a person has developed through engagement with educational practices. Competences are also defined as a set comprising theoretical knowledge, practical skills, and behaviours that enable individuals to carry out the tasks they are given, effectively and up to quality expectations [29]. Yet another view of competences is presented in [30], where they are treated as a set of behaviour patterns required for proper performance of tasks or functions.

For the purposes of present study, competences are to be understood as a set of knowledge and skills which allow an instructor to teach specific courses. From this angle, assuming that our John Smith has competences in Mathematical Analysis but does not have competences in Functions of a Complex Variable, the planner is interested in answering the following questions: Is Smith able to acquire the competences he needs to teach Functions of a Complex Variable, and if so, how long will this take and/or how much will it cost? Will Smith be able to acquire the competences he needs to teach Functions of a Complex Variable in no more than N days/weeks? Answers to this type of questions, which require operating on competences, such as comparing competences and estimating the time needed to change them, can only be obtained via mathematical modelling (using a formal mathematical approach to competence analysis). The first attempts to build mathematical competence models can be found in works [31, 32]. The competence set described in those papers includes the competences the individual employees have in a specific area. The competences are interrelated. This means that having one competence, a person can acquire another, interrelated one. A competence without which one cannot acquire a new competence is called an elementary competence [33]. Relationships between the specific elementary competences can be represented as an AND-OR graph, feature diagrams, etc. An intuitive interpretation of such relationships is presented in the digraph (AND-OR) in Fig. 4:

(a) competence D can be acquired on the basis of any elementary competence A, B or C (an (A OR B OR C) relationship),
(b) competence D can be acquired only if one has the elementary competences A and B or if one has the elementary competence C (an (A AND B) OR C) relationship).

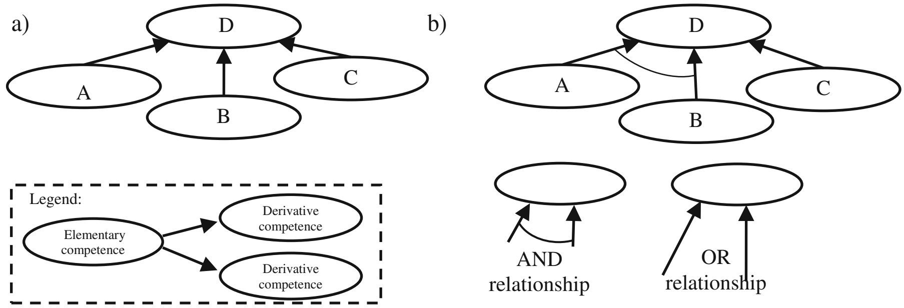


Fig. 4. Relationships among competences

Because the knowledge and skills one needs to acquire a new competence come from many different sources and can accumulate, an employee who has many different elementary competences will be able to acquire new competences sooner. This is due to the synergistic effect of the combined elementary competences, which is always stronger than the effect of individual competences. Some competence models take this phenomenon into consideration and introduce the concept of so-called compound competences to take account of the stronger elementary relationship that holds among several competences considered together as compared to the strength of the relationships considered separately [34]. In the example considered in Fig. 4a), an employee with competences A, B, and C will acquire competence D sooner than an employee with competences A and B.

In situations where competences cannot be represented as binary features, for example, when an approximate estimate of the degree to which a competence has been mastered is made, fuzzy set models are used [35]. By using various "shapes" of the membership function [36] representing the degree of membership of each competence in the set, it is possible not only to quantify the degree of mastery of the competences, but also to describe qualitatively how well they have been mastered, using linguistic values (poorly, well, to a medium degree, sufficiently well, etc.) [35, 37]. This is important when only a certain aspect of a given competence, and not the whole competence, is required to solve a given problem [34].

Authors of [13, 38] point out that employees who have the same set of competences may differ in how effectively they work. The model developed in these studies, takes into account the fact that between two employees with the same competences, the one who performs a certain activity repeatedly and continually can be more effective than the one who performs the activity intermittently. This approach uses learning and forgetting curves to determine employee performance in the production area. Such models could also be adapted in education. A good example is a situation when academic teachers more willingly/effectively teach courses that they have taught for several years in a row. When an instructor has had a break from teaching a course, they have to invest a lot of time in revising the material required for running classes in that subject area.

It seems that formal competence models can be adapted to solve TAP. In particular, they can be used to seek answers to synthesis questions about the competence structures of academic staff.

4 Remarks

The numerous examples cited from the literature show that the problem of assigning teachers to courses is a topical issue in current scientific research. First of all, it should be noted that relatively few authors consider models that take into account the competence structure of the available academic staff. On the other hand, most of the models which do pay attention to competence assume that the available staff meets the requirements resulting from the expected assignment of teachers to courses (instructional quality, no overloading with teaching duties, etc.). In practice, however, there may be a situation when the competences of the academic staff are insufficient (too few employees, employees do not have the required competences or have an insufficient degree of competence, etc.). In such situations, it is necessary to look for competence structures that will guarantee the fulfilment of expectations given the planned teacher assignment. In other words, competence management is crucial, and its aim is to ensure proper professional development of academic staff (e.g. training at a specific pace and within the given budget). To properly manage competences, one has to take account of the relationships that hold between them, such as the impact of the competences one already has on the time one needs to acquire a new competence, losing competences to acquire others, learning/forgetting, etc. The state of the art in competence modelling shows that existing approaches partially provide for this type of management, i.e. they allow one to look for answers to the question of which employee should have his/her competences raised, in what time and at what cost, to ensure that the requirements are met.

Furthermore, most research on formal competence models is focused on the processes of optimally (i.e. time- or cost-effectively) expanding the existing competence set. Most often, the purpose of these models is to minimize the cost of expanding the competences. However, there may occur a case where no changes in the structure of competences guarantee a success. This means that it becomes more important than optimization to find whether there exists a structure of competences which guarantees obtaining the permissible solution. The analysis carried out of the source literature serves to confirm the open nature of this issue.

Last observation is that, teacher assignment studies very rarely mention creating assignment plans robust to the adopted set of distractions (e.g. to employee absenteeism, changes in the number of course sections and the related need for additional staffing, etc.).

5 Conclusions

As it is evident from the literature review, teacher assignment problem is classified into the class of university scheduling problems and is most often solved before the problem of arranging the course timetable. It was proposed to extend the existing UCTTP classifications. Moreover systematized the concepts and terms used in the discussed area.

The contribution of this study is also the observation that researches related to the problem of TAP have a undeveloped area understood as their approaches don't enables the synthesis of competence structure. Additionally the lack of potential solutions to the problem and its NP-hardness nature require searching for sufficient conditions, whose fulfilment guarantees the existence of the teach assignment plan. The search for such conditions becomes particularly important – it determines the purposefulness of work- and time-consuming searches.

The next stage of research will be devoted to the development of an original model of TAP [25–27] that will take into account the constraints related to the order of acquiring new competences and the time/cost associated with obtaining new knowledge. The use of the new model will be conditioned by the application of an appropriate method of searching for competence structures. Our future research will therefore focus on reviewing the methods currently used in various approaches to the teacher assignment problem which include heuristic methods (greedy algorithms, hill climbing, etc.), meta-heuristic methods (simulated annealing, tabu search, genetic algorithms, etc.), and hybrid approaches.

References

1. Gunawan, A., Ng, K.M., Poh, K.L.: A mathematical programming model for a timetabling problem. In: World Congress in Computer Science, Computer Engineering and Applied Computing, pp. 42–47 (2006)
2. Wang, Y.Z.: An application of genetic algorithm methods for teacher assignment problems. Expert Syst. Appl. **22**(4), 295–302 (2002)
3. Zibran, M.: A multi-phase approach to university course timetabling, M.Sc. thesis (2007)
4. Ferreira, P.: Application of scheduling techniques to Teacher-Class Assignments. Dissertation submitted to obtain the Master Degree, Lisboa (2015)
5. Gunawan, A., Ng, K.M., Ong, H.L.: A genetic algorithm for the teacher assignment problem for a university in Indonesia. Int. J. Inf. Manage. Sci. **19**, 1–16 (2008)
6. Hultberg, T.H., Cardoso, D.M.: The teacher assignment problem: a special case of the fixed charge transportation problem. Eur. J. Oper. Res. **121**, 463–473 (1997)
7. Thongsanit, K.: Solving the course - classroom assignment problem for a university. Silpakorn U Sci. Tech. J. **8**(1), 46–52 (2014)
8. Domenech, B., Lusa, A.: A MILP model for the teacher assignment problem considering teacher's preferenes. Eur. J. Oper. Res. **249**(3), 1153–1160 (2016)
9. Gunawan, A., Ng, K.M., Poh, K.L.: Solving the teacher assignment-course scheduling problem by a hybrid algorithm. In: World Academy of Science, Engineering and Technology, vol. 33, pp. 259–264 (2007)
10. Hmer, A., Mouhoub, M.: Teaching assignment problem solver. In: García-Pedrajas, N., Herrera, F., Fyfe, C., Benítez, J.M., Ali, M. (eds.) IEA/AIE 2010. LNCS (LNAI), vol. 6097, pp. 298–307. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13025-0_32
11. Hosny, M.: A heuristic algorithm for solving the faculty assignment problem. J. Commun. Comput. **10**, 287–294 (2013)
12. Bettinelli, A., Cacchiani, V., Roberti, R., Toth, P.: An overview of curriculum-based course timetabling. TOP – Official J. Spanish Soc. Stat. Oper. Res. **23**(2), 313–349 (2015)
13. Małachowski, B., Korytkowski, P.: Competences-based performance model of multi-skilled workers. Comput. Ind. Eng. **91**, 165–177 (2016)
14. Małachowski, B.: Competence-based management of knowledge workers in project-oriented organizations. In: Xiong, H., Lee, W.B. (eds.) KSEM 2011. LNCS (LNAI), vol. 7091, pp. 281–292. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25975-3_25
15. De Werra, D.: An introduction to timetabling. Eur. J. Oper. Res. **16**(1), 67–77 (1985)
16. McClure, R.H., Wells, C.E.: Modelling multiple criteria in the faculty assignment problem. Socio-Econ. Plann. Serv. **21**, 389–394 (1987)
17. Qu, X., Yi, W., Wang, T., Wang, S., Xiao, L., Liu, Z.: Mixed-integer linear programming models for teaching assistant assignment and extensions. Sci. Program. **2017**, 7 (2017)
18. Moreira, J.J., Reis, L.P.: Multi-agent system for teaching service distribution with coalition formation. In: Rocha, Á., Correia, A.M., Wilson, T., Stroetmann, K.A. (eds.) Advances in Information Systems and Technologies, pp. 599–609. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36981-0_55

19. Carter, M.W., Laporte, G.: Recent developments in practical course timetabling. In: Burke, E., Carter, M. (eds.) PATAT 1997. LNCS, vol. 1408, pp. 3–19. Springer, Heidelberg (1998). https://doi.org/10.1007/BFb0055878
20. Babaei, H., Karimpour, J., Hadidi, A.: A survey of approaches for university course timetabling problem. Comput. Ind. Eng. **86**, 43–59 (2015)
21. Uney-Yuksektepe, F., Karabulut, I.: Mathematical programming approach to course-teaching assistant assignment problem. In: Proceedings of the 41st International Conference on Computers & Industrial Engineering, pp. 878–883 (2011)
22. Hamdi, K.: A Mathematical model and a GRASP metaheuristic for a faculty-course assignment problem for a university in Saudi Arabia. In: 2014 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), pp. 672–676 (2014)
23. Schniederjans, M., Kim, G.: A goal programming model to optimize departmental preference in course assignments. Comput. Oper. Res. **14**, 87–96 (1987)
24. Tillett, P.I.: An operations research approach to the assignment of teachers to courses. Socio-Econ. Plann. Sci. **9**, 101–104 (1975)
25. Szwarc, E., Bach-Dąbrowska, I.: Computer supported change management in course assignment planning process (in polish: Narzędziowo wsparte zarządzanie zmianą w procesie planowania obsady zajęć dydaktycznych), Przedsiębiorczość i Zarządzanie (2018) (in print)
26. Szwarc, E., Bach-Dąbrowska, I., Bocewicz, G.: Course assignment planning in structure of competences constraints (in polish: Planowanie obsady zajęć dydaktycznych w warunkach ograniczeń struktury kompetencji). Zielona Góra (2017) (in print)
27. Szwarc, E., Bocewicz, G.: Academic staff competences and minimum academic staff complement in course assignment planning (in polish: Kompetencje kadry akademickiej i minima kadrowe w planowaniu obsady zajęć dydaktycznych). Zeszyty Naukowe. Organizacja i Zarządzanie/ Politechnika Łódzka, Issue 69, pp. 113–128 (2017)
28. Competencies, K.: A developing concept in general compulsory education. European Commission, Eurydice, Brussels (2002)
29. Levy-Leboyer, C.: La gestion des compétences. Les Editions d'Organisation, Paris (1996)
30. Woodruffe, C.: What is meant by competency? In: Boam, R., Sparrow, P. (eds.) Designing and Achieving Competency. McGraw-Hill, New York (1992)
31. Yu, P.L., Zhang, D.: Competence set analysis for effective decision making. J. Control Theor. Adv. Technol. **5**(4), 523–547 (1989)
32. Yu, P.L., Zhang, D.: A foundation for competence set analysis. Mathe. Soc. Sci. **20**, 251–299 (1990)
33. Korytkowski, P., Malachowski, B.: Competence-based performance analysis of U-shaped assembly lines. In: Burduk, A., Mazurkiewicz, D. (eds.) ISPEM 2017. AISC, vol. 637, pp. 209–216. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-64465-3_21
34. Małachowski, B., Zaikin, O.: Competence-based human resource assignment in research projects. In: Borangiu, T., Dolgui, A., Dumitrache, I., Filip, F. (eds.) Proceedings of the 14th IFAC Symposium, Bucharest, pp. 1165–1169 (2012)
35. Wang, H., Wang, C.H.: Modelling of optimal expansion of a fuzzy competence set. Int. Trans. Oper. Res. **5**(5), 413–424 (1998)
36. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**(3), 338–353 (1965)
37. Lin, C.-M.: Multiobjective fuzzy competence set expansion problem by multistage decision-based hybrid genetic algorithms. Appl. Math. Comput. **181**(2), 1402–1416 (2006)
38. Korytkowski, P.: Competences-based performance model of multi-skilled workers with learning and forgetting. Expert Syst. Appl. **77**, 226–235 (2017)

WBT-Master - Modern Learning Management System

Nikolai Scerbakov(✉) and Frank Kappe

Institute of Interactive Systems and Data Science, Graz University of Technology, Graz, Austria
{nsherbak, fkappe}@iicm.tugraz.at

Abstract. An effective learning management system (LMS) is a vital component of an overall E-Learning infrastructure in universities nowadays. Learning Management System can be seen as a structured repository of courseware materials provided with additional communication facilities such as discussion forum, annotations, chats, etc. In this paper, we present innovative features of a modern LMS called WBT-Master. These features actually justify the development of yet another LMS.

We describe the architecture of the system, data structuring paradigm, interface to popular cloud services, social computing component and human-computer interface solutions.

We also discuss advantages of the implementation, problems that we experienced and first results of actual usage of the LMS.

The practical value of this paper is defined by possible reproduction and further development of the presented technical solutions in other LMS.

Keywords: E-Learning · Learning management system · Software architecture Cloud service · Social networks

1 Introduction

WBT-Master is an innovative Learning Management System (LMS) that was developed and successfully used at Graz University of Technology, Austria for several years [7]. There are a number of issues that justify development of yet another LMS and distinguish the new system from already existing implementations [1–3]. Thus additionally to such common for any LMS functionality as course announcements, file repositories, curriculum descriptions, file exchange, discussion forums, chats, email exchange, online quizzes and opinion polls, document annotation and evaluation facilities [2–4], etc.

- The system utilizes comparatively new architectural solutions known as Asynchronous Java and XML (AJAX) [5];
- The system supports innovative data structuring paradigms that allow successful use of the system by teachers and students that possess few or almost no knowledge on the Internet and computers [9];
- The system reuses functionality of a number of cloud services such as DropBox, Google Doc, MS One [6], etc.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 461–475, 2018.
https://doi.org/10.1007/978-3-319-99972-2_38

- The system utilizes the social network's functionality for teaching and learning [4];
- The system supports innovative human-computer interface (HCI) solutions that make the interface of the system intuitive for people having some experience with desktop computers [7];
- The system supports a number of innovative e-learning scenarios that practically implement such advanced paradigm as learning-by-doing, flipped learning, collaborative problem solving [8, 10–14], etc.

In this paper, we briefly introduce all the issues as they were implemented in WBT-Master. We also discuss advantages of the implementation, problems that we experienced and first results of actual usage of the LMS.

2 AJAX Architecture

The system is implemented as a so-called AJAX application. AJAX is a technique for accessing web servers from a client by means of special XMLHTTP JavaScript objects. AJAX-based LMS can be seen as a combination of server-side WEB services (Back-End Servlets) and Dynamic HTML (DHTML) documents that can change their appearance directly on the client without re-loading the documents from the server (Front-End). Such DHTML pages can access the WEB services and request data from the server by means of the special JavaScript object; get responses and process data on the client side. Normally, Front-End and Back-End components use XML or JSON as a communicational protocol.

The AJAX architecture has a number of well-known advantages:

- since user actions do not require re-loading HTML pages from the server, the system demonstrates very fast functionality and comfortable response time;
- since the server just implements a number of WEB services for accessing/modifying database, and all the data processing is carried out on a client side, the architecture greatly decreases the workload on the server site.

The advantages mentioned above are of great importance for modern LMS because they facilitate user satisfaction with the system functionality and performance especially in the situation where hundreds or even thousands of students work simultaneously in situations of time restrictions, for example, online examination, approaching deadlines, etc. Moreover, the actual installation of the LMS back-end does not require powerful hardware and/or special hardware solutions like server clusters, load balancers, reverse proxy, etc.

Another important aspect of this architectural solution is a possibility to use multiple front-ends for communicating to one and the same set of WEB services. In other words, there may be different user-clients that can be used in different circumstances (desktop client, tablet, smartphone, etc.) (Fig. 1).

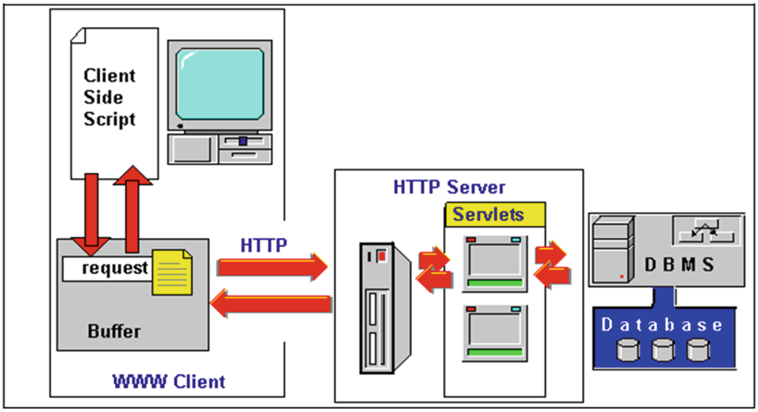


Fig. 1. AJAX architecture

Moreover, the architecture paves a new way for implementing User Interface in LMS. Traditionally, User Interface is based on the “Jump” metaphor when users click on active HTML fragments - so-called “Anchors”, and another document is visualized on the client screen. AJAX architecture allows using such modern User Interface solutions as - document event model, dynamic prompts, floating windows and virtual pop-ups.

Unfortunately, there are also a number of disadvantages of the architecture:

- the AJAX systems normally have serious restrictions in using browser “next” and “prior” buttons for navigation;
- the AJAX systems do not provide a reasonable level of security by default, and special preconditions must be taken.

The AJAX navigational problem results from the typical AJAX situation when users perform some actions, the document on the screen is dynamically changed, but URL of the document stays the same. In this case, usage of the “back” button does not recover a previous picture on the client screen but jumps to another HTML document having different URL. The solution is rather simple, but requires additional efforts from the system developers - they must change URL in the address bar for each AJAX action, for example, by adding “#[number]” notation to the original URL.

AJAX security problems may be seen on two levels. The Back-end component of an AJAX system can be seen as a number of WEB services that can be invoked by means of an HTTP request from the front-end component. Hence, the services potentially can be invoked by any Internet client that can mock the request format. This problem can be efficiently solved by dynamically generated tokens that identify authorized clients. Thus, we use two different schemas for the token generation:

- as a particular client is authorized, the system generates a token that is kept on the server as a session variable and on a client as a JavaScript variable. Any further requests are additionally authorized with this token that is checked on the server before fulfilling the request;

- the above method is not secure enough since the token value is valid for a number of requests. In cases that require a higher level of security, tokens are generated for each request.

The second level of the security leak is explained by the main AJAX principle where software (JavaScript functions) are fetched from the server and evaluated on the client side that makes altering and manipulation with the scripts possible. Actually, the problem of manipulation with client-side scripts does not have an ultimate solution in AJAX. The only possible remedy is double checking the most sensitive transactions on the server site. For example, if a WEB service that returns info on a particular student by ID is invoked, the service may check whether the client has really got rights to do so by checking user-name and privileges on the server session.

3 Data Structuring Paradigm

WBT-Master is a Learning Management system that is used to create so-called e-learning courses by teachers. Each e-learning course is a structured repository of online materials accompanied with additional communications facilities. Structuring online data repositories is not a trivial task and often requires additional knowledge from the teachers' perspective. WBT-Master provides an advanced data structuring paradigm that can be used by teachers without any additional knowledge and skills.

In WBT-Master, a particular e-learning course is structured as a number of data containers belonging to different types. The number and types of containers reflect a particular training scenario for the training course. Each container encapsulates a number of elements and provides a particular browsing strategy that allows access to the elements

- The "Announcements" container is a list of members; normally members are ordinary HTML or text documents. All the announcements are visualized as a scrollable list of members.
- The "Course Curriculum" container is a set of textual or other documents that are visualized as a pull-down menu of members. Normally, these documents describe schedule, goals, prerequisites and requirements for the course.
- The "Course Library" container is a collection of training documents that are visualized as a tree-like structure consisting of folders and sub-folders.
- The "Teaching Aids" container is an object similar to the "Course Library" object, but elements of such "Teaching Aids" container are other applications. For example, "Quiz", "User Uploads", etc. are data objects.

The teacher may define an arbitrary number of containers of different types; each container can be accessed as a particular tab on the top and visualized as defined by its type (Fig. 2 shows a course consisting of four containers and visualization one particular container - "Course Library" in this particular case).

For example, a user may click on the Tab "Course Library" (see Fig. 2) to access individual documents of the course by means of browsing tree-like structure.

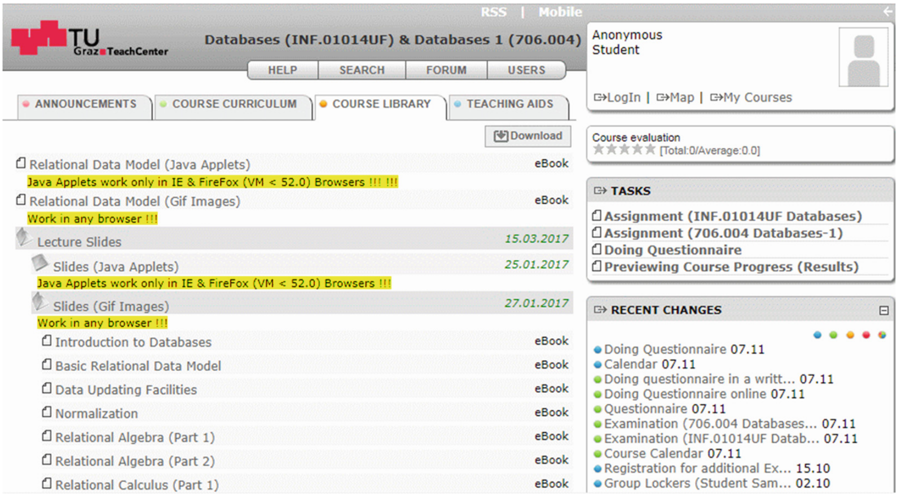


Fig. 2. WBT-Master e-learning course

Additionally to the previously mentioned data containers, the system supports a big number of purpose-oriented applications such as “Quiz”, “User Uploads”, “Chat”, “Discussion Forum”, “E-Book”, etc.

Building an online training course typically follows a top-down approach.

- First, a particular course is created and gets an id, title and general properties (open date, close date, access restrictions, etc.). Thus, the course is an “empty” container that can host a number of functional components.
- Second, the teacher defines a set of the course components needed to implement the desired training scenario. In the simplest case, an “Announcements”, “Course Curriculum” and “Course Library” containers are sufficient. Obviously, teachers may use more containers such as “Teaching Aids”, and more elements such as “Discussion Forum”, “Chat”, “Quiz”, “Opinion Poll”, etc. to implement the desired training scenario. All the containers that are needed for the course are created as objects of a particular type and inserted into the “Course” container. For example, the teacher can create an empty “Announcements”, an empty “Course Curriculum” and empty “Course Library” containers thus creating a template for a new training course.
- Finally, multimedia page, textual documents and/or movies are created. They then become elements of previously defined containers as above.

Once a training course has been created, elements can be inserted, modified or removed in a very flexible way and without any knowledge of Internet and programming. Insertion of a new element into the “Announcement” object, for example, is carried out by means of such “usual” user interface solutions as drag-drop, prompt menus, etc. Such insert action automatically updates the list of all course announcements. Conversely, removing a member will remove an entry from the list. Similar, adding and deletion of files in the course library will modify the content of folders from the user perspective. With the course library, of course, we must also specify the insertion position (target folder) in the hierarchy.

Please note that the navigable structure is separated from rather than embedded in the course elements. This is, of course, essential since a document or application can be re-used in another course.

4 Cloud Services

Modern cloud services are an important e-learning resource nowadays. The services can be seen from three perspectives:

1. The services provide a reliable repository of files accessible via the Internet, the repository is available anytime and everywhere;
2. The services provide powerful tools that can be used to create and edit files directly on the Internet. The files can be automatically converted into a number of common formats;
3. The services are accomplished with an especially developed client that integrates the service into an operating system on a client computer. Thus, the operations with remote files can be done exactly in the same way as operations with local files.

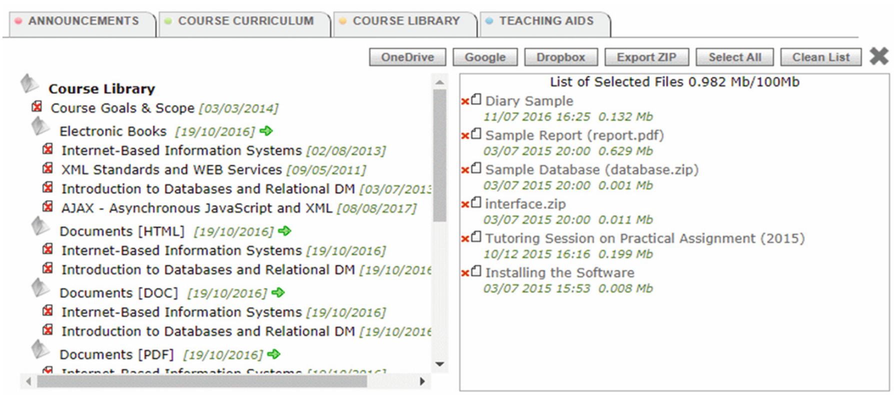


Fig. 3. Cloud services in WBT-Master

In WBT-Master, cloud services are reused in a number of different ways. Thus, individual cloud accounts can be registered on WBT-Master and the students can copy files from the LMS into a cloud service repository with one mouse click (see Fig. 3). Since the cloud services are seamlessly integrated with a local file system, this process can be seen as an automatic delivery of files from LMS to a local desktop.

Obviously, the possibility to copy a particular file from the LMS onto a local file system via the cloud service does not give crucial advantages over conventional downloading files via the browser.

The method can be further extended using the concept of synchronization. The students can create directories on the cloud service repository and synchronize the directories with the folders on the course library, announcements and course curriculum.

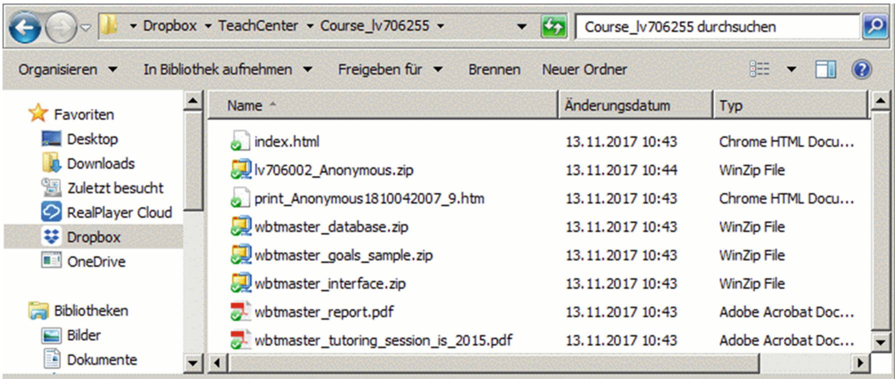


Fig. 4. Synchronization WBT-master and dropbox

Thus, the cloud service provides an access from the local file system to the latest versions of the files uploaded onto WBT-Master. In this case, students do not need to browse the course Internet sites at all, the latest versions of the files are automatically delivered to their desktops from the selected folders (see Fig. 4).

Similarly, the teacher can define particular directories to be synchronized with a training course on WBT-Master. In this way, teachers may simply edit files in such directories locally, and the files are automatically uploaded to the e-learning course via the cloud service. Moreover, the files can be edited using the advanced tools provided by the cloud service, and automatically upgraded in the course repository. For example, the teacher may define the course announcements as textual files in the special Google Doc repository, and edit the files using the powerful and user-friendly Google online editor. The announcements are automatically uploaded onto the e-learning course as a part of the synchronization procedure.

Other important aspects of any LMS are facilities for online collaborative authoring of textual documents that are often needed in e-learning. Since many cloud services allow users to edit textual documents in collaboration, the scenario is implemented in WBT-Master on the base of such cloud services. There is a special WBT-Master add-on component called “Group Lockers”. This component supports the collaborative authoring of text documents, spreadsheets and presentations by a group of students. Students or teachers create so-called “Group Lockers”. A “Group Locker” is a named memory space where students can create their files. The files are synchronized with the files in the cloud repository. Individual files may be evaluated by a teacher, and the results are visible for students. Thus, students may write articles, press releases, to-do lists, etc., in collaboration with group members all working on the same document at the same time via the cloud service.

5 Social Network Functionality

User management and course management sub-systems of WBT-Master are extended with a rather comprehensive social functionality. As users access the system they see a list of students and teachers that are currently online. The list can be filtered using a number of parameters – users working on the same course, users subscribed for the same study curriculum, etc. The users can exchange with instant messages and initiate a multi-user chat session.

A particular user account is visualized for other users as a so-called User Identity Card (see Fig. 5). Basically, the User Identity Card just provides an info on the user – name, affiliation, location, E-Mail, etc. Additionally, each user account is provided with a number of functional components such as:

- "Friends" is a list of other users that are linked to the current user in some sense. The term "linked" deserves an additional explanation: users may explicitly declare another user as a friend; additionally, students may enroll one and the same course and the enrollments can be done in the same or different years. All the situations above are weighted with a number of points. Suppose, two students declared each other as friends and enrolled a number of one and the same courses. In this case, the link between such students is rather strong - has got more points. In other words, each user is related to a number of other users with a number of links having different weights. The links can be sorted and filtered using the weight parameter. The networks of users interconnected by means of links can be browsed to access a particular User Identity Card.
- "Pinwand" provides a possibility to publish series of messages (Blog) that belong to this particular User Identity Card.
- "Applications" allows activating references to external applications. For example, references to a particular blog, Jaiku, Tweeter or email account can be activated and inserted into this particular User Identity Card. In this case, the information provided on the card is extended with recent Twitter messages, recent Blog posts and similar.
- "Locations" implements the concept of so-called Social Bookmarking. Social Bookmarking services allow users to store links to private files, posts and external web resources in a structured and easily accessible way. The links can be created, and modified by the owner of the User Identity Card. The bookmarks can also be accessed, evaluated and annotated by other students.
- "Albums" and "Files" provide an access to personal repositories consisting of private and public folders. Pictures and other files can be uploaded to the folders by the owner of this Identity Card.

Since the functionality of User Identity Cards is very similar to user profiles on social networks, the User Identity Cards can be optionally synchronized with the user accounts on Facebook. In this case, any messages, photos and files uploaded to Facebook are automatically copied onto WBT-Master and vice versa.

One of the important components of any social network is creating and subscription for a user groups. In WBT-Master such user groups are created automatically. All the courses on the server are provided with so-called Course Identity Cards. The structure of the Course Identity Card is almost identical to the previously described structure of the User Identity Card (see Fig. 5). The Course Identity Card is connected to all students (User Identity Cards) that were enrolled for the course. Students enrolled for the course have a full access to the Course Identity Cards; the students can upload files and pictures, write posts to the course blog ("Pinwand") and define bookmarks that they find relevant to the course content.

User Identity Cards, Course Identity Cards and the "friends" and "enrollment" relationships form the LMS social network.

Another social dimension of the content is provided by the evaluation facilities that allow students to comment and evaluate any contributions done by other users. Thus, each contribution gets an average evaluation scope that reflects a user satisfaction with this resource.

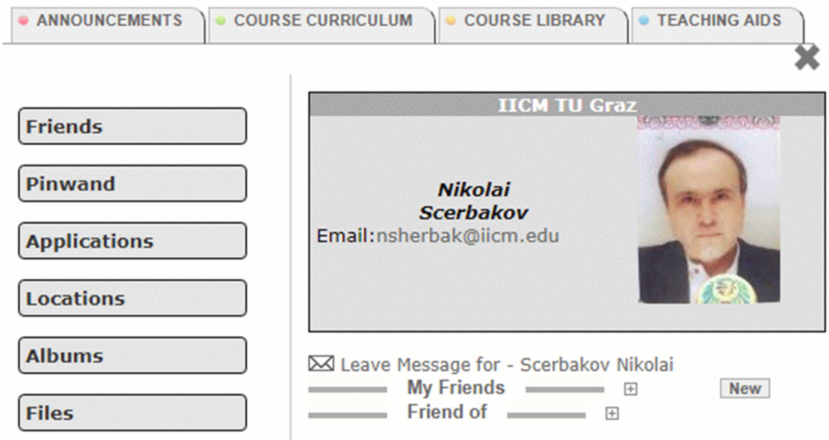


Fig. 5. User identity card in WBT-master

The LMS social network can be used in e-learning on two levels. First, the network can be seen as an additional "teacher-independent" educational level where students can exchange with sample assignments and examination questions, share an experience with different courses, share their course notes, recommend internet resources, etc.

Since the components of the Course Identity Card (posts, files, pictures, bookmarks, etc.) can be potentially created by any student enrolled for the course, and the Course Identity Cards are persistent over years, we can expect rather big numbers of such contributions for each e-learning course, and this is hard to know which resources are relevant. Hence, successful usage of the student's contributions by other students requires a kind of recommendation system. WBT-Master utilizes the idea of collaborative filtering. Simply stated, the method is based on two types of data:

- the similarity of users;
- satisfaction of users with particular resources.

and on the following precondition: "If the user A is similar to the user B, and the user B was satisfied with a particular resource, we expect that the resource is of interest to the user A as well".

As a particular student accesses the collection of shared course contributions (Course Identity Card), all the evaluation values are multiplied by the weight of the link between the student and author of the evaluation. Such new modified evaluation figures are used to range the bookmarks in accordance with expected preferences of this particular user.

6 Human-Computer Interface Solutions

Historically, web-based and local desktop applications use different Human-Computer Interface (HCI) solutions.

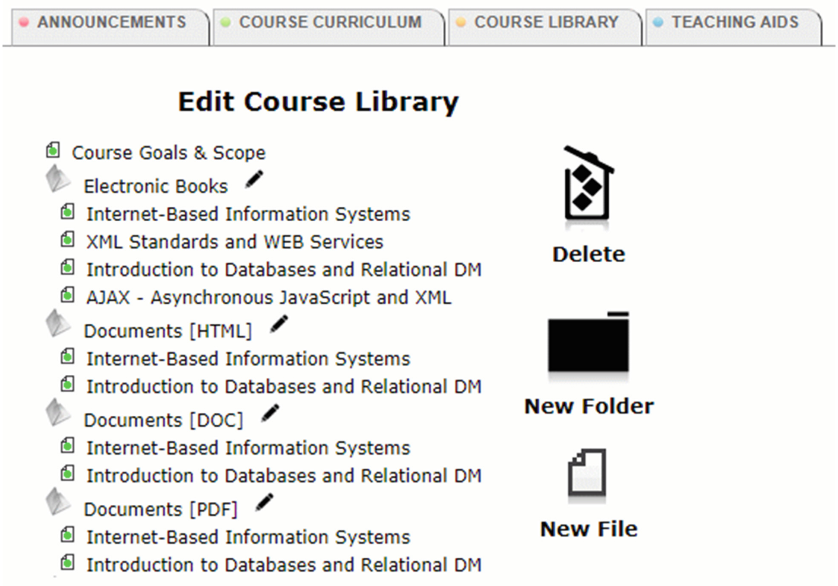


Fig. 6. Editing a course library

WEB-Based applications normally utilize such HTML specific HCI elements as clickable anchors and HTML forms. Desktop applications go much further in this respect; and actively use event model where prompts may appear as cursor-over events, contextual menus pop-up with a right mouse click, elements can be dragged-dropped. Another important aspect of HCI is responsiveness - the specific ability to respond to the user actions in a timely fashion. Responsiveness is rather different to support for web-based applications since HCI actions in such systems require client-server communication.

Users are used to advanced interface solutions, and only HCI built in accordance with commonly accepted principles of desktop applications can be considered powerful and intuitive nowadays. AJAX architecture and HTML5 extensions form a solid basis for providing WEB-Based applications with advanced and intuitive HCI.

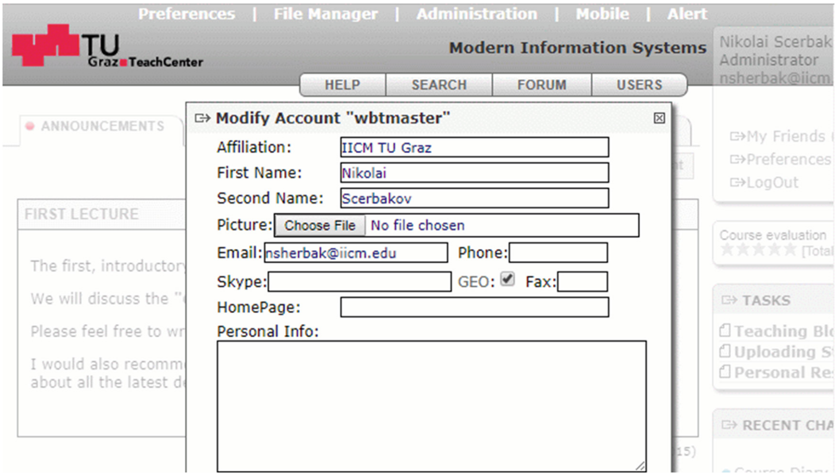


Fig. 7. Secondary virtual pop-up for an optional task

Thus, for example, WBT-Master utilizes the following advanced features of HTML5:

- Drag and Drop interface for implementing of data editing actions (see Fig. 6 where all the files can be Dragged and Dropped to implement the advanced editing facilities);
- Context Menus are available as individual elements on the screen as the cursor is moved over the element;
- All the user actions that potentially can cause a substantial delay (for example, uploading big files) are provided with progress indicators;
- Optional functionality is available in secondary virtual pop-ups (See Fig. 7);
- Graphical perception of the user interface elements is significantly improved by using vector graphics (HTML5 Canvas elements) and even animations.

7 Innovative E-Learning Scenarios

Generally speaking, an e-learning course can be seen as a combination of at least three components:

- structured repository of training resources;
- tools to provide communication and support by the teacher and other students;
- set of tasks that students must complete.

The Sects. 3 and 4 above present WBT-Master as an online platform for flexible delivery training materials to students. Section 5 presents some unusual ways of collaboration as students work with e-learning courses. The third component of e-learning course - tasks to be completed by students is supported in WBT-Master via a number of so-called Plug-In applications. The applications implement different training scenario. The teacher simply selects some applications and places them into the “Teaching Aids” container. WBT-Master currently allows selecting from a list of over 30 applications. In this paper, we just describe a couple of the Plug-In applications to illustrate such innovative educational paradigms as learning-by-doing and flipped classroom learning [8–13].

Learning-by-doing education is implemented by means of the special “Project” application. The application operates with a number of so-called “Projects”; each “Project” is a protected memory space that can be used by a group of students - project members. “Projects” are supposed to be created by teachers. The teacher defines a title, short description and a maximum number of project members for each project. There is a special procedure for registration of students for a project. Consequently, any student enrolled for a particular course, may select a project and register as a project member while there is still a free place.

Essentially, “Projects” incorporate files of two different types, there are so-called teacher files that can be uploaded only by teachers, and cannot be modified or deleted by the project members. Normally, such teacher files contain information necessary for the students to develop the project. Project members work with another type of files that are called project files. Registered project members may access teacher files and project files; they may upload, modify and delete project files.

To provide a necessary level of collaboration and support, project files:

- may be commented by project members and/or by the teacher;
- may be evaluated by a teacher, and results are visible for project members.

There is also a special messaging mechanism where e-mails may be broadcasted to members of a working group.

Flipped classroom principles are implemented with another so-called “Problem-Solving” application that will be called “Problems” for short. In this case, teachers also create individual working spaces that are called “Problems”. There is also a working group of students associated with each “Problem”. Students may enroll themselves in a working group similar to the previous case. There are three types of files that can be uploaded into the “Problem” memory space:

- teacher files that define a problem to be solved;
- technology description files that identify a particular technology and described it on a level sufficient for solving the problem;
- project description files that simply illustrate the project implementation stages similar to the previous case of “Projects” (Fig. 8).

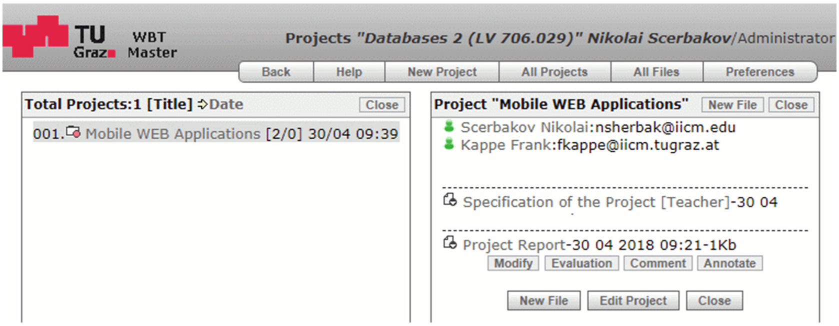


Fig. 8. The “Project” application

Thus, the scenario gets started with the definition of problems by the teacher. The teacher is not supposed to define any technological solutions or recommendations; the teacher just defines a goal of the whole group work. For example, a typical problem definition may look as follows: “Develop a mobile application working with a camera on Samsung Note mobile device”.

The working group members must choose a set of relevant technological solutions sufficient for achieving the goal defined by the teacher. Then the students describe the existing technology on a level sufficient for solving the problem. This task is reported on the server in the form of the technology description files. For example, there are a number of solutions for the problem above, but the group members can decide to implement the project by means of HTML5, CSS3, JavaScript and Cordova framework. In this case, the working group develops textual files describing the selected technologies and publishes the files on the server. Essentially, other students working on other problems may access the technology description files, comment and evaluate them; and, thus, learn those technological solutions.

Finally, the working group develops the application and publishes the report on the server where it is finally evaluated by the teacher to assess the students’ progress with the course.

8 Conclusion

In summary, the following features of WBT-Master distinguish it from other existing Learning Management Systems:

- The system makes use of comparatively new architectural solutions and internet technologies (See Sect. 2);
- The system offers a new data structuring paradigm that goes beyond structuring of the repository by means of hypermedia links (See Sect. 3). Using the editable containers as the main paradigm for data structuring allows successful use of the system by teachers and students that possess few or almost no knowledge on the Internet and computers;
- The system makes use of functionality of a number of cloud services (See Sect. 4). The concept of synchronization of the repository of online training resources with a cloud service implements a concept of delivering materials to a student's desktop and offline editing of e-learning content by teachers.
- The system utilizes sound principles of social computing (See Sect. 5), and supports concepts of the "teacher-independent" level of e-learning, and implements a recommendation engine for e-learning resources.
- The system supports innovative human-computer interface (HCI) solutions (See Sect. 6). Hence, the interface to the system looks familiar and intuitively understandable for people having some experience with desktop computers;
- The system supports a number of innovative e-learning scenarios that practically implement such advanced paradigm as learning-by-doing, flipped learning, collaborative problem solving, etc.

References

1. Report – LMS Trends 2015: Is it time for something different?https://www.docebo.com/blog/report-lms-trends-2015-brandon-hall/. Accessed 1 May 2018
2. Ramani, R.: Preview: 11 Learning Management System Technology Trends by The In-dustry's Top Analysts. https://elearningindustry.com/11-learning-management-system-technology-trends-top-analysts-2017-preview. Accessed 1 May 2018
3. Uzunboylu, H., Bicen, H., Cavus, N.: Efficient virtual learning environment: case study of Web 2.0 tools and Windows Live Spaces. Comput. Educ. **56**, 720–726 (2011)
4. Dobre, I.: Learning management systems for higher education - an overview of available options for higher education organizations. Proc. Soc. Behav. Sci. **180**, 313–320 (2015)
5. Vivekan, S.K.O., Mistry, M.: Impact of AJAX in web applications. Int. J. Adv. Eng. Technol. **3**(1), 144–145 (2012)
6. Scerbakov, A., Ebner, M., Scerbakov, N.: Using cloud services in a modern learning management system. J. Comput. Inf. Technol. **23**(1), 75–86 (2015)
7. Scerbakov, N.: TU Graz WBT-Master. http://coronet.iicm.tugraz.at/wbtmaster/welcome.html. Accessed 1 May 2018
8. Kappe, F., Scerbakov, N.: File uploading scenarios in a modern learning management system. In: Proceedings of 9th International Conference on Education and New Learning Technologies, Barcelona, Spain, 3rd–5th July 2017, pp. 4904–4909 (2017)

9. Kappe, F., Scerbakov, N.: Object-oriented architecture of a modern learning management system. In: Proceedings of 9th International Conference on Education and New Learning Technologies, Barcelona, Spain, 3rd–5th July 2017, pp. 4910–4916 (2017)
10. Scerbakov, N., Kappe, F.: "Course Diary" - an innovative e-learning scenario. In: Proceedings of 12th International Technology, Education and Development Conference, Valencia, Spain, 5th–7th March 2018, pp. 2823–2828 (2018)
11. Scerbakov, N., Kappe, F.: Visual programming of e-learning scenarios. In: Proceedings of 12th International Technology, Education and Development Conference, Valencia, Spain, 5th–7th March 2018, pp. 2861–2867 (2018)
12. Scerbakov, N., Kappe, F., Schukin, A.: Social bookmarking as a component of e-learning. In: Proceedings of in: World Conference EdMedia + Innovate Learning 2018, Amsterdam, Netherlands, 25th–29th June 2018. Accepted for publication
13. Scerbakov, N., Kappe, F., Pak, V.: Collaborative document authoring as an e-learning component. In: Proceedings of in: World Conference EdMedia + Innovate Learning 2018, Amsterdam, Netherlands, 25th–29th June 2018. Accepted for publication
14. Scerbakov, N., Kappe, F., Schukin, A.: "Mentoring Session" - an e-learning scenario based on reverse AJAX architecture. In: Proceedings of 10th International Conference on Education and New Learning Technologies, Palma de Mallorca, Spain, 2nd–4th July 2018. Accepted for publication

The Mobile Application Based on Augmented Reality for Learning STEM Subjects

Tomas Valatkevičius(✉), Andrius Paulauskas, Tomas Blažauskas, and Reda Bartkutė

Kaunas University of Technology, Kaunas, Lithuania
{tomas.valatkevicius,andrius.paulauskas, tomas.blazauskas,reda.bartkute}@ktu.lt

Abstract. App store is full of programs, which are based on augmented reality. A lot of studies showed that augmented reality has a lot of benefits for user's ability to learn new things or their increase in motivation. However, the majority of these programs are dedicated for entertainment and just several of them are designed for learning processes. The authors have developed an app based on an augmented reality, which aims to provide scientific formulas for math, physics and chemistry subjects and by this to ease the exercise solving. The target group is K12 learners in school. The app was uploaded to the app store for both, iOS and Android operating systems. Further researches have to be done on finding the impact of this type of programs. If the impact would be positive, the authors suggest improving an app by adding more specific topics.

Keywords: Augmented reality · Mobile application · STEM · Learning

1 Introduction

In Lithuania, the education system is quite static and lacks different learning methods and styles. Students are not involved in a learning process and it reflects in the way how they generate information in the future. The Stanford Teaching Commons indicate that active learning is meaningful not only for a child but for a student also. It indicates that when students are engaged in active learning, they are more fascinated with learning a subject and learn more, whether it is a practice, discussion or some application [1].

A not new term in the field of technology is augmented reality. It incorporates virtual reality into the real world and is interactive [2]. Apart from its advantages, AR is not widely used in an educational process, in Lithuania. Another issue here is that a lot of teachers do not imagine how VR and AR technologies can be incorporated into learning processes. One of the simplest ways to do it is a mobile application for learning [3].

The use of mobile applications for studying in Lithuanian schools and universities is still not so popular. The old methods used, without any teaching interactions and the advantages of the new emerging technologies remain untouched. Another problem with the usage of mobile learning applications related to the fact that there are quite a small amount of educational content applications in the market, especially when talking about STEM subjects.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 476–482, 2018.
https://doi.org/10.1007/978-3-319-99972-2_39

2 Methodology

First of all, the literature on the augmented reality-based mobile applications models used for education in the world was overviewed. In the second part of the article, the mobile application "KTU Decode" was introduced and briefly presented. For the research part, were taken two focus groups to evaluate the effectiveness of the "KTU Decode" application. It was done by evaluating the results while learning with not AR or VR related applications and learning with the "KTU Decode". Results were briefly presented in the diagrams and described. All in all, the authors have made the conclusions with some recommendations and future perspectives.

3 Literature Review

When people hear an augmented reality, it mostly is associated with the use of games and fun experiences. However, in the whole world, there are a thousand good practices of augmented reality usage in the educational process. The cheapest way for all students to have interactive and emerging new technologies in their everyday learning process is mobile applications based on augmented reality [4]. These applications are usually free and require just a smartphone and the camera [5].

In 2014, two IT developers from Malaysia have created an augmented reality-based mobile application for learning science subject in primary schools. Their application called iSains and had two topics: moon phases, day, and night. The application let students see different moon phases in 3D through their textbooks. The authors had claimed that further researches have to be done to know the exact effect on pupils learning process and more topics should involve making this augmented reality-based application more useful [6].

Research on an augmented reality-based mobile application for learning made in 2014, in Taiwan. The research took the place at natural science course in elementary school. The authors have constructed an augmented reality-based learning system, which was created by using these tools: JAVA, Oracle, Xcode. The operating principle of the system based on the GPS location of the students and it guided them to the ecology areas where they could do the special tasks. Pupils could use the camera to capture the images of the real environment and later on students can edit images, according to the tasks and share them with their community through the WIFI. In the test presented feedbacks from 57 pupils. Results have revealed that students showed higher motivation in learning, lesson attendance and confidence [7].

In 2015, Bernadette Perry from Canada created and implemented a prototype of an "Explorez", which is an augmented reality-based mobile learning tool to help students to learn the French language. This prototype implemented into a first-year University French-language class. The author asked students about their experience using this learning-tool and all of them were positive. Majority of the students have identified that it was a great experience and it was very nice to get out of a class, even just through the mobile phones. They liked the concept that everything is going on in a real environment but with some added augmented reality elements. Perry has identified that there should be further researches on the feedbacks after a long time but for this day increased students motivation to learn is clearly seen [8].

The benefits of augmented reality are obvious and easily acceptable. Without the benefits listed above, there are much more. For example, in engineering graphics lessons, augmented reality book used for students to see different objects from 3D perspectives and to solve complementary problems. The questions and problems defined in that way, that students would be able to train their mental rotation abilities. After the test, researchers claimed that augmented reality-based learning has a positive effect on students' spatial abilities improvement [9].

Other researches about augmented reality-based mobile learning applications revealed that this type of tools increase collaborative learning [10], motivation to perform learning activities [11], improve learning attitude and learning outcome [12], enrich ideas [13] and engage concurrence and satisfaction [14].

After an overview of the previous researches, there can be claimed that mobile applications based on VR or AR are widely used and contributes to better study results. However, in Lithuania, such examples are not seen and not a lot of researches were done to see if this type of learning may be included in Lithuanian educational system.

4 Overview of "KTU DECODE"

"KTU Decode" is an augmented reality-based mobile application, which purpose is to complement the presence of the Kaunas University of Technology: agendas for math physics and chemistry. With the help of this app, in the agendas, users can see virtual formulas. In every different agenda, users can see formulas for the specific wanted theme. The application reflects virtual formulas for specific themes and with the help of it, it is easier to solve exercises.

The target group of the application is higher classes for learners at school. The aim of the app is to ease the solving of exercises by using formulas for math, physics and chemistry.

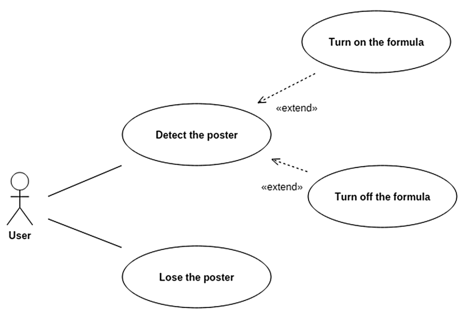


Fig. 1. Use case diagram.

In the figure above (Fig. 1) you can see the use case diagram. In the program one user engaged, that can do several tasks. The entire task is listed in the figure and described below.

Poster detection. To detect the poster it is necessary to turn on the app and to allow it to use your device camera. After the camera directed to the poster, the app detects what kind of poster it is and formulas appear in the screen.

Turning on the formulas. Each of the posters has different themes and formulas that are complimenting one specific poster. A user is able to choose themes and by pressing on them to turn on the layer of scientific formulas, that can be switched.

Turning off the formulas. A user, which already turned on the layer of formulas, by pressing on the same theme twice, can turn off an additional layer of formulas.

Lose of the poster. When the poster is lost, a user cannot turn on or turn off the scientific formulas.

5 The Model of "KTU Decode"

The app is based on two main components: "Unity 3D" game engine and "Vuforia" the library of an augmented reality platform. The logic of an app was written in C# language. In the figure below (Fig. 2) is the model of the app.

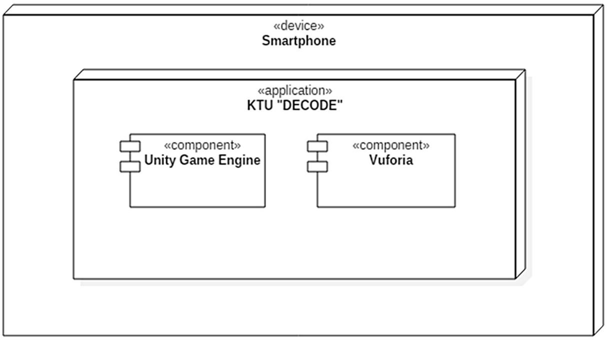


Fig. 2. Deployment diagram of the app.

The process is starting with the "Vuforia" which uses a camera of the smartphone to locate points of the interests. The points transferred to Unity game engine in which a content of an augmented reality is drawn.

For the use of "Vuforia", the unique app licensee identification is the must. This identification generated in "Vuforia" management interface by registering the app in progress. The license identification must be transferred to "Vuforia" configurable file in "Unity 3D" game engine.

The app of "KTU Decode" does not use in "Vuforia" platform provided, points of interests, based on cloud computing. As points for an app, posters were used (Fig. 3). The worksheets of the posters were uploaded to the "Vuforia" database through its management interface.

Fig. 3. Screen view of the app in the mobile.

The system processes these posters and gives the quality rate as a feedback. The quality rate is the explanation of how good the points of interests will be followed. Processed databases of images were downloaded and implemented into the project.

The app was uploaded into an “App store” and “Google Play” stores and is free to download. When uploading the app to “iOS” operational system, “Unity 3D” tools have been used for generating “XCode” environment.

6 The Experiment on the Mobile Application Effectiveness

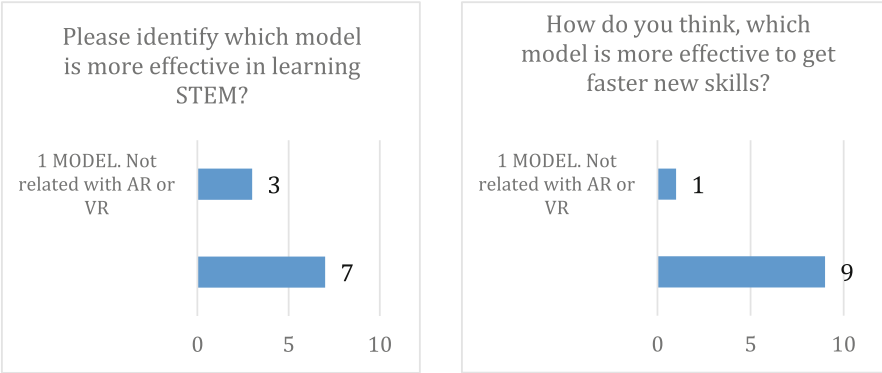
To evaluate the effectiveness of the mobile application taken two aspects: existing applications for educational needs and learning results.

There do exist many applications for learning, but there are not so many related to VR and AR. For the experiment, there were taken two types of applications 1st model (not related with AR or VR) and the 2nd model related with AR and VR (see Fig. 4).

To evaluate “KTU Decode” were invited experts, who provided their opinion about the models’ integration into practice.

For the experiment, there were taken two different models to evaluate the effectiveness of the KTU Decode application. There was created the questionnaire for the experts to compare that two methods of implementation. The 1st group used 1 METHOD, the 2nd group used 2 METHOD.

Finally, we can declare that the 2nd model is more useful in learning STEM subjects. Totally 7 respondents declared the functionality of the 2 model is more effective in a learning process and totally 9 experts declared that this model is more effective to get new skills.



a) Evaluation of effectiveness learning in STEM.

b) Evaluation on the effectiveness to get new skills by the model.

Fig. 4. Expert's evaluation results.

7 Conclusions

As the great number of studies showed that an augmented reality can help students to remember everything better and to engage in the learning process, the authors have decided to try to create the simple mobile application, which would be based on an augmented reality. The augmented reality-based mobile application created to help students to solve exercises for math, physics and chemistry faster and easier. The authors have uploaded the app into the app stores for both, iOS and Android operating systems. "KTU Decode" is free to access and use by students.

However, no studies were done to investigate the benefits of the app: if it is useful or how to improve it; if the app help to solve exercises faster or not. The future research can be done by investigating the impact of this kind of programs for students. For the app improvement, the authors suggest adding more topics/themes.

References

1. Teaching Commons Homepage. https://teachingcommons.stanford.edu/. Accessed 30 Mar 2018
2. Barfield, W. (ed.): Fundamentals of Wearable Computers and Augmented Reality. CRC Press (2015)
3. Ledford, C.J., Canzona, M.R., Cafferty, L.A., Hodge, J.A.: Mobile application as a prenatal education and engagement tool: a randomized controlled pilot. Patient Educ. Couns. **99**(4), 578–582 (2016)
4. Rauschnabel, P.A., Brem, A., Ro, Y.: Augmented reality smart glasses: definition, conceptual insights, and managerial importance. Unpublished Working Paper, The University of Michigan-Dearborn, College of Business (2015)
5. Page, T.: Application-based mobile devices in design education. Int. J. Mob. Learn. Organ. **8** (2), 96–111 (2014)

6. Majid, N.A.A., Husain, N.K.: Mobile learning application based on augmented reality for science subject: Isains. ARPN J. Eng. Appl. Sci. **9**(9), 1455–1460 (2014)
7. Chiang, T.H., Yang, S.J., Hwang, G.J.: An augmented reality-based mobile learning system to improve students' learning achievements and motivations in natural science inquiry activities. J. Educ. Technol. Soc. **17**(4), 352 (2014)
8. Perry, B.: Gamifying French language learning: a case study examining a quest-based, augmented reality mobile learning-tool. Procedia-Soc. Behav. Sci. **174**, 2308–2315 (2015)
9. Hung, Y.H., Chen, C.H., Huang, S.W.: Applying augmented reality to enhance learning: a study of different teaching materials. J. Comput. Assist. Learn. **33**(3), 252–266 (2017)
10. Ke, F., Hsu, Y.C.: Mobile augmented-reality artifact creation as a component of mobile computer-supported collaborative learning. Internet High. Educ. **26**, 33–41 (2015)
11. Ibáñez, M.B., Di Serio, Á., Villarán, D., Delgado-Kloos, C.: The acceptance of learning augmented reality environments: a case study. In: 2016 IEEE 16th International Conference on Advanced Learning Technologies (ICALT), pp. 307–311. IEEE, July 2016
12. Cai, S., Chiang, F.K., Sun, Y., Lin, C., Lee, J.J.: Applications of augmented reality-based natural interactive learning in magnetic field instruction. Interact. Learn. Environ. **25**(6), 778–791 (2017)
13. Wang, Y.H.: Exploring the effectiveness of integrating augmented reality-based materials to support writing activities. Comput. Educ. **113**, 162–176 (2017)
14. Wu, P.H., Hwang, G.J., Yang, M.L., Chen, C.H.: Impacts of integrating the repertory grid into an augmented reality-based learning design on students' learning achievements, cognitive load and degree of satisfaction. Interact. Learn. Environ. **26**(2), 221–234 (2018)
15. Faster pedagogical framework for steam education based on educational robotics. https://doi.org/10.14419/ijet.v7i2.28.12897

The Ways of Using Augmented Reality in Education

Daina Gudonienė(✉) and Tomas Blažauskas

Kaunas University of Technology, Kaunas, Lithuania
{daina.gudoniene, tomas.blazauskas}@ktu.lt

Abstract. As the improvement in the technologies is fast and widespread, it should be used for its best. Emerging technologies like an augmented reality are used to play games when at the same time it can be the best motivating educational tool. However, augmented reality tools are not spread in schools or higher education institutions, just several apps are designed for the educational purposes. The problem is that there is no one effective model which would help to increase the efficiency of the learning process and the gap between the need of different types of using augmented reality (AR) in education and different subjects is still existing. The authors provide the model of AR for the effective learning processes.

Keywords: Augmented reality tools · Education · Learning process
Effective learning · Mobile applications

1 Introduction

Virtual reality, augmented reality, robots, technology-enhanced learning, mobile learning, serious games and other terms are seen and heard more usually than before. That is because we are living in the technology age. Emerging technologies now are at the peak and inventors are trying to integrate these technologies in our everyday life as fast as it is possible. Not an exception is education [1]. More and more researches are focused on the benefits of new technologies when providing educational services. In particular, when talking about augmented reality, researches are divided into two parts: mobile augmented reality devices [2, 11], and non-mobile augmented reality devices [3, 12].

Augmented reality itself, merges two components: real life and virtual objects [13]. AR technologies have improved so much that now it can stimulate not only sense of vision but also smell, touch, taste and hearing [4, 14]. An augmented reality now are changing teaching activities and engaging students in learning. The most common and way for students and teachers are mobile applications based on augmented reality. It is easy to use and do not require some special technologies. The user just has to have a smartphone with a camera, WIFI and GPS sometimes. However, if you would look through the mobile apps stores, the applications for education took smaller part than those for entertainment. From this follows that all possibilities of AR are untapped. AR is not widely used in education and this is a problem - there are no models provided to educators in different subject areas.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 483–490, 2018.
https://doi.org/10.1007/978-3-319-99972-2_40

In this review, the authors have analyzed the applications based on augmented reality because this type of augmented reality is mostly used for education and learning. It will cover marker, markerless based and location-based, augmented reality tools for education.

Akçayır and Akçayır [1] identified the advantages of AR, which are related to students' learning outcomes such as learning achievement, motivation, and attitude are gathered under the learner outcomes category. Also, the reported challenges involve application-related and technical problems. Most of the technical problems were experienced in location-based AR applications. The identified challenges are mostly related to the design process and will be not reviewed in the paper.

2 Methodology

The article goes as follows: in the first part of the paper, there are identified the problem and the need for the research area. Literature overview of augmented reality-based tools for education is presented. The discussion of the research is carried out by presenting new ways to implement AR in education. Experimental evaluation of the new method is presented and the effects showed in the paper. Conclusions provided at the end of the paper by making an overview on the paper and showing the effectiveness of an augmented reality.

3 Literature Review

Some researches were done to analyze if the mobile applications based on augmented reality has an impact on learning progress and the whole education system. Education here represents education in general: from elementary school to vocational education. Some researches were done to analyze if the mobile applications based on augmented reality has an impact on learning progress and the whole education system. Education here represents education in general: from elementary school to vocational education.

In 2015, the research on augmented reality based application impact for VET students was made. The application, called "Pain-cAR" was introduced to help students in VET sector that are learning about the process of repairing paint on the car. Students in VET sector lack of motivation, concentration, confidence, etc. The application was created with the purpose to increase the interest in learning subject and to overcome some barriers. In the experiment were introduced 13 students from VET sector (all were male). Results on four dimensions (attention, relevance, confidence, satisfaction) were positive. It means that augmented reality-based mobile application had a positive impact on students in VET sector [5].

HuMAR is an augmented reality based application, which can be used in Android tablets to help to understand the structure of bones of a lower appendicular skeleton. Later on, more features were added to enhance the system of any selected bones group. In the experiment were engaged, 30 students. After the feedback, results have been conducted and conclusions were made. The authors revealed the problem that this kind of applications are not used in education widely, even if those applications help to remember information longer and increase student's motivation to learn [6].

Shirazi and Behzadan have determined the problem of decreasing number of students in STEM subjects. The authors have made an assumption that students are not interested in the learning methodologies and that is why they do not apply for STEM subjects. Before the experiment, the questionnaire for engineering students was made and the results revealed that students in engineering subjects are not interested in their learning methods and there is a lack of practice. According to the results, the authors predicted that new technologies in educational process would be not only more interesting for students but at the same time will get them a lot of benefits. Two groups of engineering students were taken: one group was working with AR tools (CAM-ART) and another without AR. After the experiment, results shown that students are more engaged to work with CAM-ART and students have shown better results when working with new technologies than without it. The application was based on building and calculating the accuracy of buildings [7].

The application for 6th-grade students to get them interested in biology and environmental science by interacting them with the natural environment – EcoMOBILE was presented in 2013 [8]. The application was based on location trigger. EcoMOBILE was created to recognize plants and to help to students to learn in the future by saving all information into databases. Also, the tasks were prepared, according to the information they had. Later on, this program was used for students of botanic in the university [9]. The results of both alternatives showed that students were more interacted with the environment, exploration was more interesting, and learning became easier.

'The Water Drop's Adventure' is an app, designed on marker-based augmented reality. It helps for pupils to learn about the marine world. Talking more precisely, fishes. The augmented reality markers are used to identify different learning objects (fishes). First of all, the interactive story is told with the promotion of the fish characteristics. In the process of the storytelling, pupils are answering questions through somatosensory interaction. It is increasing acceptance of the study material. Later on, this app was upgraded with several different games, based on the same principle, just with different subjects. Interaction with the virtual environment was observed as improving and engaging [10].

The researches [7–10] analyzed above, covered up these types of augmented reality for possible educational learning objects or activities in education (Table 1).

The authors have identified that all of these types of augmented reality have a positive impact on students' motivation and the whole educational process.

4 Discussion About the Model for Learning Objects Design Based on AR

Visualization of the learning content based on reality it is an effective learning method. The authors suggest the model of the augmented reality-based learning object what helps to students' in effective way to get new skills (Fig. 1).

Table 1. Types of AR.

Type	Description	Visualization
Marker-based	Virtual reality objects are projected onto the screen based on real-life marker position	
Marker less	Virtual reality objects are projected onto the screen based on real-life object position	
Location-based	It uses GPS or other location tools in the user's smartphone and may provide data for restaurants, shops, parks and etc., based on the users' location	

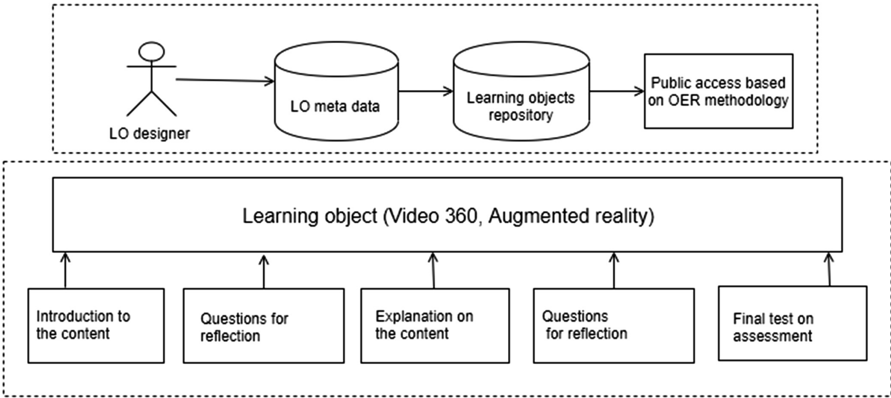
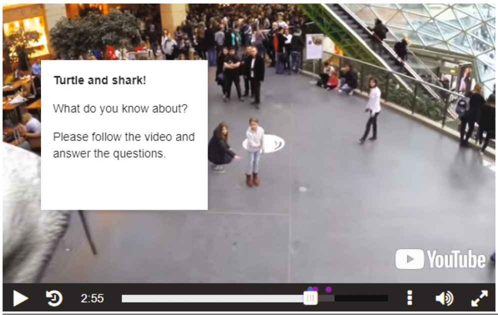
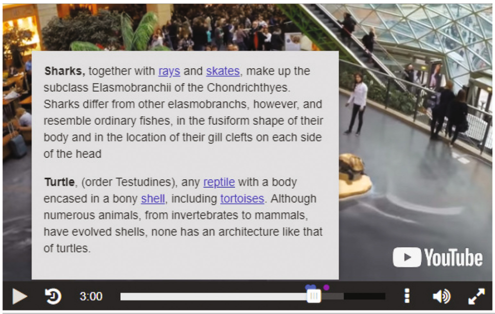
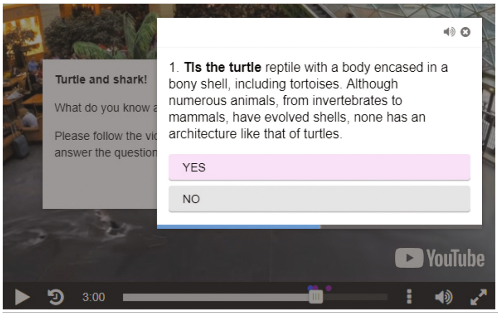
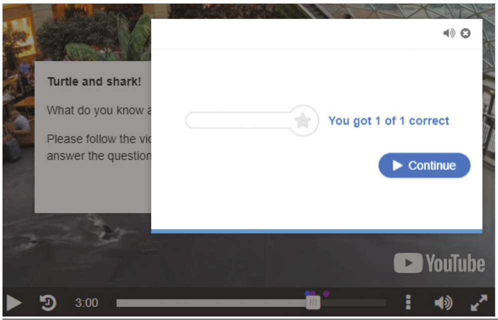


Fig. 1. The model of the LO design based on AR.

According to the presented model (Fig. 1), there was designed LO. The new method of the using AR in the practice can offer a new way to improve educational practices, to extend the use of LO and to reduce the time required to prepare technologies enhanced learning (Table 2).

Table 2. The functionality of the AR based model.

<table>
<tr><td>
</td><td>Presentation of the learning object</td></tr>
<tr><td>
</td><td>Providing the text information to the video-based information</td></tr>
<tr><td>
</td><td>Providing assessment tool to learners</td></tr>
<tr><td>
</td><td>Getting feedback on the results of achievements by following correct answers</td></tr>
</table>

5 Experimental Evaluation

The authors designed the learning objects based on virtual reality and asked to professional working in IT to evaluate the effectiveness of the learning object design process (see the object with the link http://adunooc.ndma.lt/en/node/233/).

Totally 10 experts evaluated the design process and the effectiveness of providing information to students (Fig. 2).

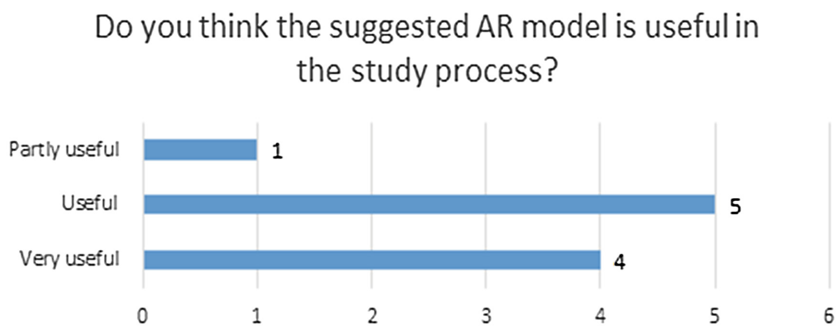


Fig. 2. The using AR model in study process.

The new ways of learning become more effective, not in schools but in the universities as well. The augmented reality provides to users a wide opportunity to have a real view on the subject, within the classrooms or in distance way with very low efforts and finances just having a link and internet (Fig. 3).

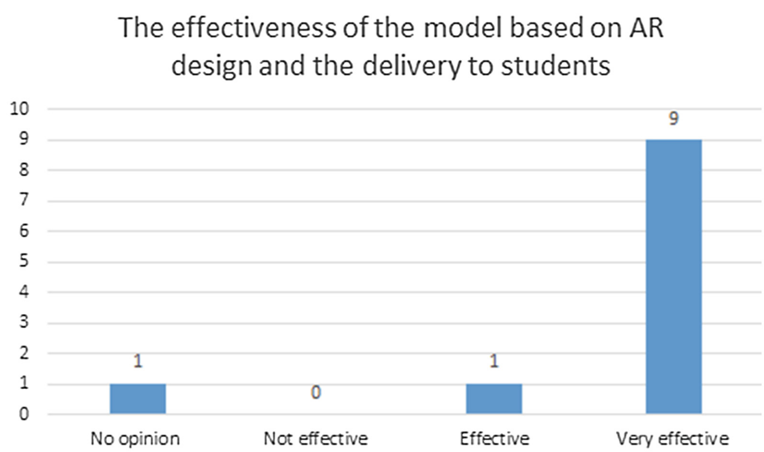


Fig. 3. The effectiveness of the model.

There is many research on the interactive learning objects and its effectiveness. The tool h5P provides the opportunity to LO designers to make the video more interactive by integrating different explanations and to use different types of questions for the skills assessment making the study process more effective and more interesting to students.

6 Conclusions

Augmented Reality can also be used to enhance collaborative tasks. It is possible to develop innovative computer interfaces that merge virtual and real worlds to enhance face-to-face and remote collaboration.

Augmented reality interfaces offer seamless interaction between the real and virtual worlds, positively influence learning, increase learners motivation and learning efficiency, AR can impact spatial visualization abilities, AR reduces misinterpretations during learning.

AR increases overall understanding of subjects. The learning objects based on AR could be integrated also to another educational context and are effective for delivery in the study process.

References

1. Akçayır, M., Akçayır, G.: Advantages and challenges associated with augmented reality for education: a systematic review of the literature. Educ. Res. Rev. **20**, 1–11 (2017)
2. Alakärppä, I., Jaakkola, E., Väyrynen, J., Häkkilä, J.: Using nature elements in mobile AR for education with children. In: Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services, p. 41. ACM, September 2017
3. Kamarainen, A., Metcalf, S., Grotzer, T., Browne, A., Mazzuca, D., Shane, T., Dede, C.: EcoMOBILE: integrating augmented reality and probeware with environmental education field trips. Comput. Educ. **68**, 545–556 (2013)
4. Arth, C., Grasset, R., Gruber, L., Langlotz, T., Mulloni, R., Schmalstieg, D., Wagner, D.: The history of mobile augmented reality developments in mobile ar over the last almost 50 years (2015)
5. Bacca, J., Baldiris, S., Fabregat, R., Graf, S.: Mobile augmented reality in vocational education and training. Procedia Comput. Sci. **75**, 49–58 (2015)
6. Jamali, S.S., Shiratuddin, M.F., Wong, K.W., Oskam, C.L.: Utilising mobile-augmented reality for learning human anatomy. Procedia-Soc. Behav. Sci. **197**, 659–668 (2015)
7. Johnson, L., Adams Becker, S., Estrada, V., Freeman, A.: Horizon report 2014 - Higher education edition. The New Media Consortium, Austin, TX (2014)
8. Lu, S.J., Liu, Y.C.: Integrating augmented reality technology to enhance children's learning in marine education. Environ. Educ. Res. **21**(4), 525–554 (2015)
9. Rohmer, K., Jendersie, J., Grosch, T.: Natural environment illumination: coherent interactive augmented reality for mobile and non-mobile devices. IEEE Trans. Visual Comput. Graph. **23**(11), 2474–2484 (2017)
10. Sardo, J.D.P., et al.: Portable device for touch, taste and smell sensations in augmented reality experiences. In: Mortal, A., et al. (eds.) INCREaSE 2017, pp. 305–320. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-70272-8_26

11. Barfield, W., (ed.): Fundamentals of Wearable Computers and Augmented Reality. CRC Press (2015)
12. Büschel, W., Mitschick, A., Dachselt, R.: Here and now: reality-based information retrieval: perspective paper. In: Proceedings of the 2018 Conference on Human Information Interaction & Retrieval, pp. 171–180. ACM, March 2018
13. Martín-Gutiérrez, J., Fabiani, P., Benesova, W., Meneses, M.D., Mora, C.E.: Augmented reality to promote collaborative and autonomous learning in higher education. Comput. Hum. Behav. **51**, 752–761 (2015)
14. Obrist, M., Velasco, C., Vi, C.T., Ranasinghe, N., Israr, A., Cheok, A.D., Gopalakrishnakone, P.: Touch, taste, & smell user interfaces: the future of multisensory HCI. In: Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems, pp. 3285–3292. ACM, May 2016

Experience with Distance Learning of Informatics

Rostislav Fojtik(✉)

University of Ostrava, Ostrava, Czech Republic
rostislav.fojtik@osu.cz

Abstract. Information and communication technologies are developing very rapidly. Learning of informatics must respond to this rapid development. Also, training of ICT staff requires flexibility and acceptance of new facts. There is a need to educate and expand your knowledge and skills continually. The traditional system of full-time education is often only suitable for young students. Those who are already employed prefer distance learning to a combined form of education. This article illustrates twenty years of distance learning experience in Bachelor and Master of Science in Informatics. Assumptions confirm that distance learning is more demanding for students than full-time learning. Distance learning requires greater motivation of students and the ability to work with time self-study. The article shows the comparison of the results and the success rate of attendance and distance students and also describes the possibilities of improving the quality of distance learning. Long-term findings show that distant students have significantly lower scores in their first years of study than full-time bachelor's students. In the following years, the differences diminish, and the student results are sometimes comparable.

Keywords: Distance learning · Education · E-learning · Informatics Learning management system · T-test

1 Introduction

Distant and blended forms of teaching are increasingly required. This is due to the need to frequently increase and supplement staff qualifications. Employees usually have to increase their qualifications and change their focus more than they did in the past. Traditional full-time teaching can often not be used in employee training. The problem of distance learning is that most participants have no practical experience with this form of teaching. This leads to worse results for distance students. Teachers need not only to prepare quality teaching materials but also to adapt their teaching methods [8, 9, 17]. It turns that distance learning is difficult for the students. The most significant problem lies in abilities of students to manage their studies. Part of students is unable to plan a well-proportioned schedule for learning. They often start learning at the end of the semester before course exam. Distance students do not have enough time to handle all the requirements, and they fail in their studies [2, 6, 10]. The first semester of education is the most problematic period of studies. Many such students interrupt their studies

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 491–501, 2018.
https://doi.org/10.1007/978-3-319-99972-2_41

during this period. One of the reasons is wrong information and incorrect idea about the distance learning [7, 16].

Within 20 years at our department, different technologies and methodologies have been tested to improve the distance and blended forms of teaching. When introducing distance learning, there was a need for teachers to undertake distance learning courses. None of the teachers had experience with this form of teaching. In the early years, Lotus Learning Space was used to install Lotus Notes Client to create and manage courses. The system was later replaced by the IBM Workplace Collaborative Learning commercial product. However, this system was designed for the corporate sphere. IBM Lotus Workplace Collaborative Learning Authoring Tool was delivered to this LMS. The application was unnecessarily complicated for most teachers and did not allow some functionality, such as randomly generating test questions and answers. That is why the system was replaced by Tutor 2000 and ToolBook Instructor, which better responded to didactic requirements. Its disadvantage was that the courses created were not simple websites and required the presence of specific plugins. Therefore, all of these systems have been replaced by LMS Moodle, which has proved to be the most appropriate system for teaching at the university. The course takes 13 weeks and ends with a test or an exam. Students have 2 to 3 tutorials during each semester. These tutorials serve as a full-time tutorial with the teacher. Distant students must also take the exam as well as full-time students at the university. Both groups of students must have the same knowledge and skills to complete courses.

2 Comparing the Success of Distance and Full-Time Students

We have tracked the results in individual subjects over the next few years. The following tables and graphs show the comparison of the success of the students in the selected courses. Already from the beginning of the implementation of distance learning, it has been shown that distance students have more difficulty in completing courses [1, 5].

The comparison aimed to find out whether distance students are less successful than full-time students and in which periods of study, distant students are significantly worse than full-time students. We have set the following zero and alternative hypotheses:

Zero hypothesis: Distant and full-time students will not have different learning outcomes.

Alternative hypothesis: Distant and full-time students will have different learning outcomes.

We compared the results of the students in individual courses in bachelor and master study. The full-time and distance form of teaching differs only in the way of teaching. Knowledge and skills requirements are the same in both forms of teaching. Students in both groups must have the same knowledge and skills to complete the courses. We used statistical methods. For comparison, the results of the two groups (distance and full-time students), we used the T-test: Two-Sample Assuming Equal Variances. Two sample t-test is used to compare the difference between two populations. This parametric test assumes that the variances are the same in both groups. This assumption we tested by the F-Test Two-Sample for Variances. For analysis of the results in each item was measured as having detected data variability. We used a coefficient of variation. To interpret the results of the second stage classification was done the t-test. For evaluating the results were used MS Excel and statistical software Wizard for the operating system Mac OS X and statistical software Statistics Visualizer for iPad [3, 4, 14, 15].

2.1 Comparison Results

Distance learning puts students at much higher demands than studying full-time. It is reflected in the results of the students. We can observe the differences between the group of distance students and a group of full-time students. The difference is mainly in the first semester. The success rate of full-time students is 64% in the first semester. The success of distance learners is only 39% in the first semester.

Figure 1 shows the comparison of the success of distance and full-time students in the course of Architecture of hardware and Fundamentals of operating systems. This course is usually studied in the first semester of a bachelor study. The graph clearly shows the worse results of distance students. The average success rate over the last five years was 70% for full-time students and 47% for distance students. A smaller percentage of successful distance students is also because students do not have a distance learning experience and they cannot correctly schedule study time. A large number of students do not study in the first semester and find out at the end of the semester that they are not able to handle the tasks and finish their studies. Many distance students do not get to the test. Comparing the average score earnings for successful students shows almost the same results. Successful full-time students earned an average of 70 points. Successful distance students received an average of 71 points. The average score for all attendance students was 51 points, and the average of all distance students was only 30 points. Number of students: 2013 - number of full-time students = 119 / number of distance students = 55, 2014–111/79, 2015–90/54, 2016–75/57, 2017–63/57.

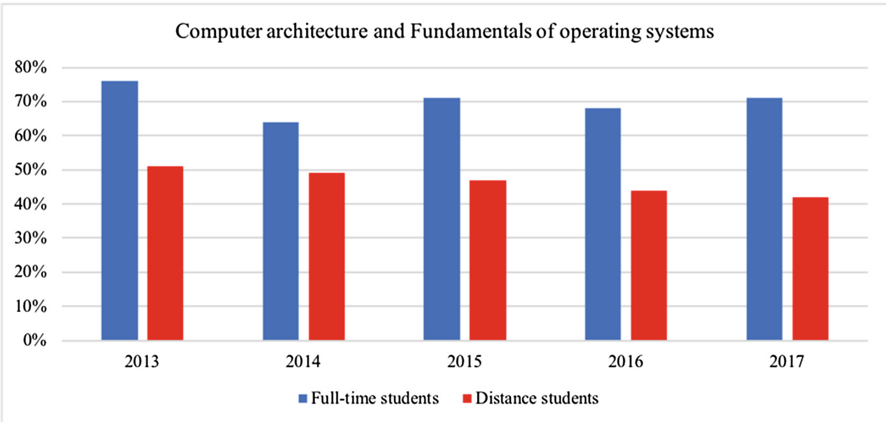


Fig. 1. Comparing success in the course Architecture of hardware and Fundamentals of operating systems (Typically in the first semester).

Table 1 shows a comparison of students' success in the course Architecture of hardware and Fundamentals of operating systems. The results show that we can reject the zero hypothesis. The results confirm that distant students have worse outcomes than full-time students (Fig. 2).

Table 1. T-Test: Two-Sample Assuming Equal Variances for course Architecture of hardware and Fundamentals of operating systems (year 2017).

t-Test: Two-Sample Assuming Equal Variance	Full-time students	Distance students
Mean (average points)	50,39682	29,91228
Variance	1149,88837	1317,474310
Observations (number of students)	63	57
Results		
P(T <=t) one-tail	0,000894	
t Critical one-tail	1,657869	
P(T <=t) two-tail	0,001788	
t Critical two-tail	1,980272	

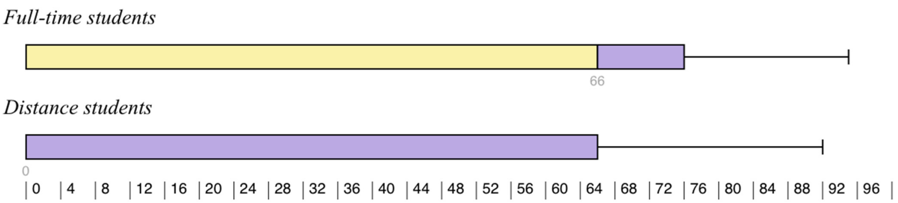


Fig. 2. Comparing success in the course Architecture of hardware and Fundamentals of operating systems (Typically in the first semester).

Table 2 compares the achievements of successful students. The results show that we cannot reject the zero hypothesis. The results of the students who have completed the course are comparable in both groups. It is confirmed that the biggest problem for distance learners is the fact that many do not start or study in the course before the final

Table 2. T-Test: Two-Sample Assuming Equal Variances for course Architecture of hardware and Fundamentals of operating systems. Only successful students. (year 2017)

t-Test: Two-Sample Assuming Equal Variance	Full-time students	Distance students
Mean (average points)	70,55555	71,04116
Variance	165,66161	158,82427
Observations (number of students)	45	24
Results		
P(T <=t) one-tail	0,44041	
t Critical one-tail	1,66791	
P(T <=t) two-tail	0,88082	
t Critical two-tail	1,99600	

evaluation. Similar results were also observed in the other courses in the bachelor's and master's studies in computer science.

Similar results were found in the Basic of programming program, which students usually also write in the first semester (see Table 3 and Fig. 3).

Table 3. T-Test: Two-Sample Assuming Equal Variances for course Basic of programming (year 2017).

t-Test: Two-Sample Assuming Equal Variance	Full-time students	Distance students
Mean	44,125	25,66666
Variance	944,94772	1155,61025
Observations (number of students)	56	66
Results		
P(T <=t) one-tail	0,001126	
t Critical one-tail	1,657650	
P(T <=t) two-tail	0,002251	
t Critical two-tail	1,979930	

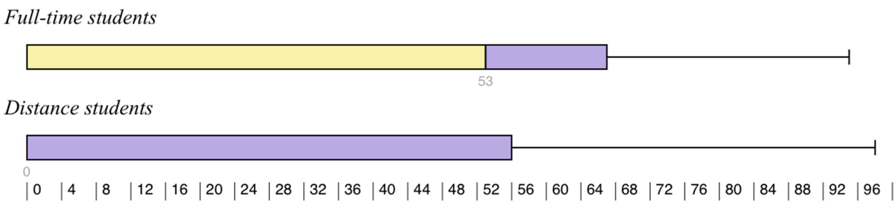


Fig. 3. Comparing success in the course Basic of programming (Typically in the first semester).

Many students finish their studies after the first semester. The difference between distance and full-time students is not so pronounced in the second semester (see Figs. 4 and 5).

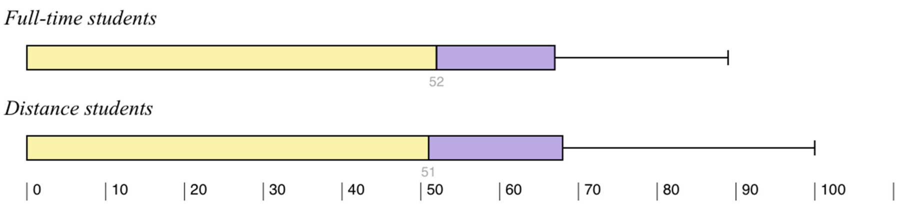


Fig. 4. Comparing success in the course Operating Systems 1 (Typically in the second semester).

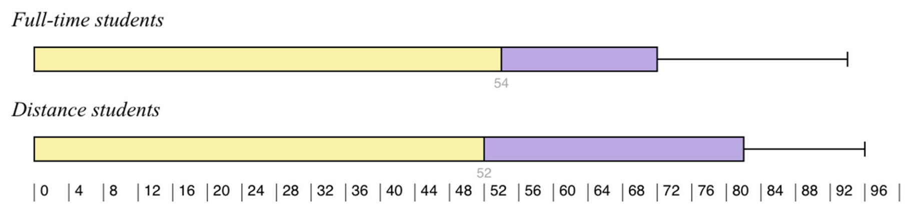


Fig. 5. Comparing success in the course Algorithms and Data Structures (Typically in the second semester).

Figure 6 shows the percentage success rate in the Programming in C/C++ subject, which is usually studied in the third semester. The average attendance rate for full-time students was 75%. Distant students had an average rate of 69%. Number of students: 2013 - number of full-time students = 40 /number of distance students = 23, 2014–39/26, 2015–36/18, 2016–15/13, 2017–28/12.

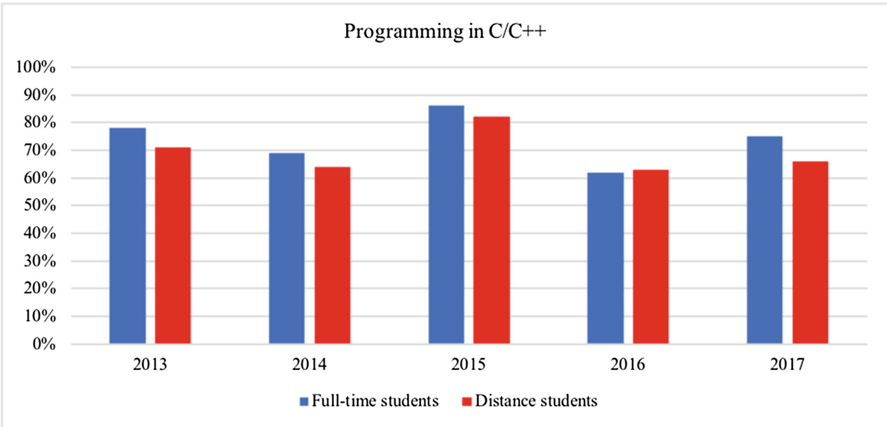


Fig. 6. Comparing success in the course Programming in C/C++ (Typically in the third semester).

Figure 7 shows the success rate in the Programming server applications subject which students are usually studying in the fifth semester. The average attendance rate for full-time students was 79%. Distant students had an average success rate of 68%. Number of students: 2013 - number of full-time students = 39 /number of distance students = 31, 2014–39/24, 2015–26/4, 2016–31/17, 2017–30/8.

The graph in Fig. 8 shows the average of successful students in some selected bachelor's degree courses in informatics. The average attendance rate for full-time students was 76%. Distant students had an average rate of 54%. List of courses typically in the first semester: APZOS - Architecture of hardware and Fundamentals of operating systems (number of full-time students = 63 /number of distance students = 57), ALDS1 – Basic of Programming (56/66). Second semester: OPSY1 – Operating

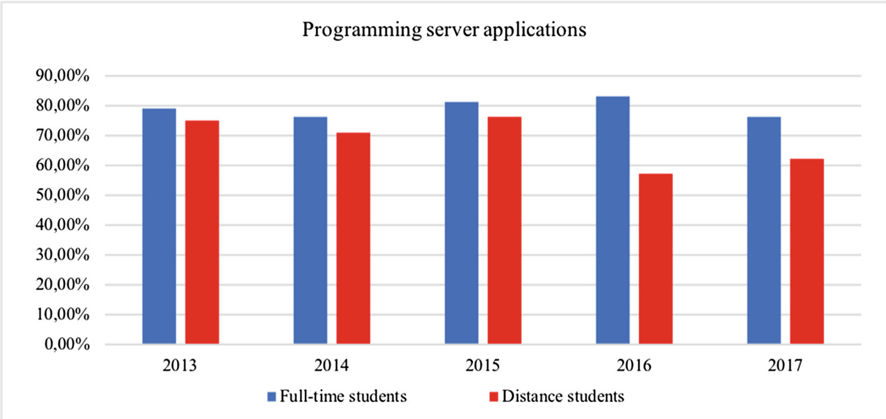


Fig. 7. Comparing success in the course Programming server applications (Typically in the fifth semester).

Systems (43/29), ALDS2 - Algorithms and Data Structures (30/15). Third semester: PRGCC – Programming in C/C++ (28/12), UVDAT - Introduction to Databases (38/19), LZUI1 - Logic for informatics (38/17), GALPR - The Principles and the Algorithms in the Computer Graphics (32/10). Fifth semester: OOPR3 - Programming server applications (30/8), SOFCO - Basics of softcomputing (32/10), ZMATS - Basics of Mathematical Statistics (28/12). Sixth semester: KKDAT - Data coding and compression (35/14).

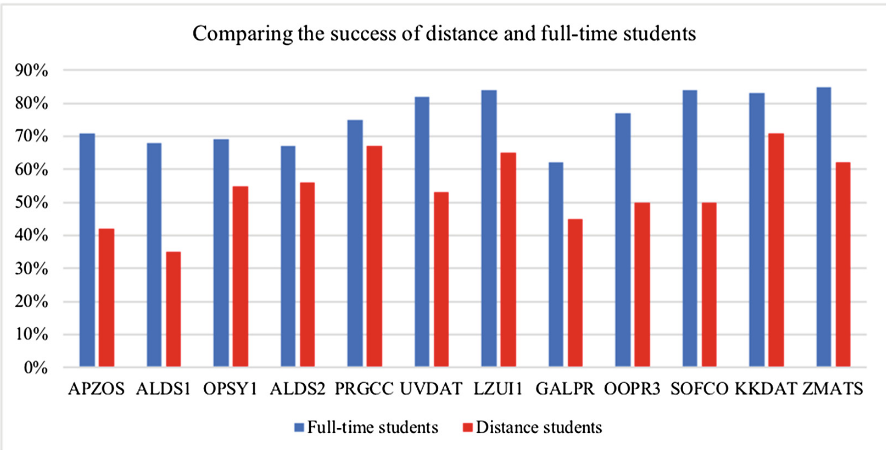


Fig. 8. Comparing success in the courses (year 2017).

The chart in Fig. 9 shows the average earnings of all students who enrolled in these subjects. The graph and results of t-tests confirm the worse results of distance students.

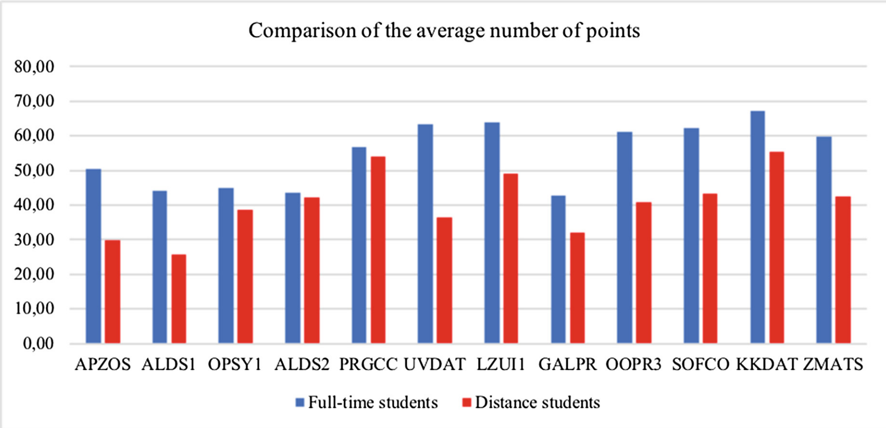


Fig. 9. Comparison of the average of points (all students).

Another comparison shows the results of only students who have completed the course. Figure 10 shows that the results of both groups are very similar. Distant students gained more points in some courses.

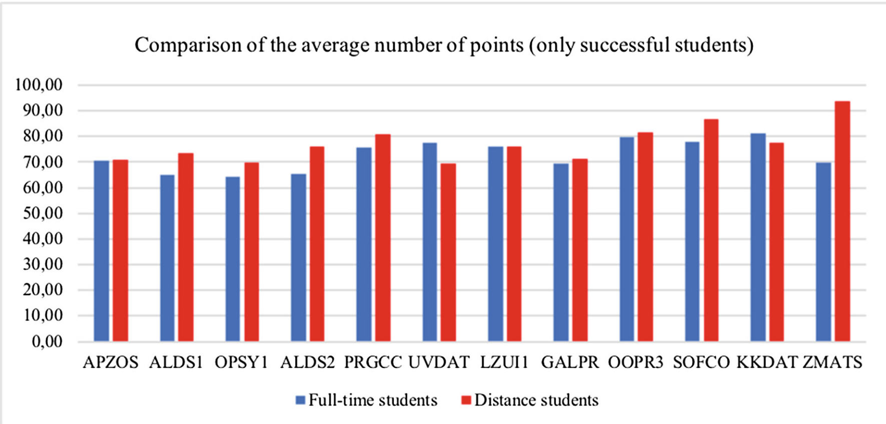


Fig. 10. Comparison of the average number of points (only successful students).

2.2 Implementation of Distance Courses

The results of the comparison show that there is a need to devote significant effort to quality distance learning. Creators of online courses must create not only high-quality teaching materials but also ensure regular communication and study of all students. The absence of direct contacts among students and teachers often leads to irregular self-study of distance students. The use of video lectures to help students with self-study problems and learning texts has proved to be beneficial. By improving the quality of teaching materials and learning itself, the success of distance students has improved but is still lower than that of full-time students. It is especially noticeable at the beginning of the Bachelor's degree program.

To improve the quality of teaching, the study materials were gradually developed. In addition to teaching texts and presentations, video recordings of lectures and seminars and other multimedia teaching materials were gradually created. MediaSite was used to record video lectures in which the video captured by the teacher synchronised with the slides of the presentation. Distant students appreciate the opportunity to play video from lectures, seminars or tutorials. Video helps them better understand learning themes.

Experience with distance course management shows that it is essential to motivate students to work actively on a continuous basis. Students who work continuously during the semester usually have better results. Continuous activity can be assured, for example, by assisting with regular tasks that students have to develop, for example, once every 2–3 weeks. A more extended period between correspondence tasks proved to be inappropriate. Students were only studying at a glance, and many of them have dropped out of their studies early. Because they left their assignments for the credit week, they found that they were not able to handle everything. A significant role is played by distance learning in communication. Due to the minimal presence of face to face communication, it is necessary to support and develop other communication tools. It was very advantageous to evaluate the activity in the electronic conference of the course. Students asked and helped each other with ambiguity. Online communication tools have been used less, due to the different time capabilities of course participants [11–13].

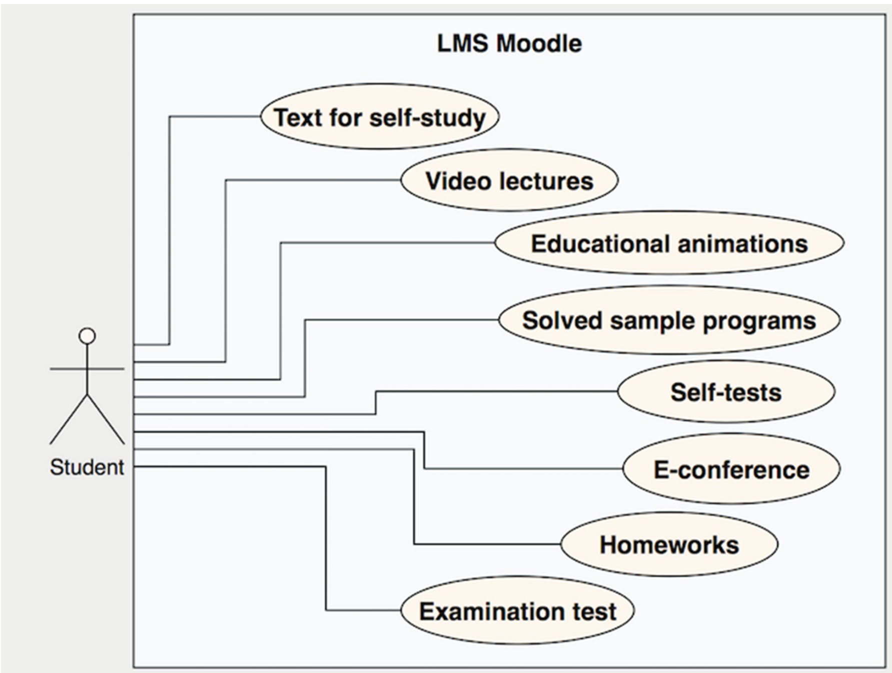


Fig. 11. Usual course content.

Figure 11 shows the essentials that should have a good online learning course. Distance courses must also ensure that students acquire not only knowledge but also practical skills. For example, the ability to generate codes and to program. Therefore, instructional videos and demonstrations of concrete examples are significant [18].

3 Conclusion

Distance learning is popular today, and it allows students to study at work or a great distance. This fact corresponds to the requirements for frequent change of professional knowledge and skills. Distance learning in conjunction with information and communication technologies brings new opportunities and benefits. At the same time, however, this form of teaching brings new problems. The research results show that distance students have worse outcomes than full-time students. Students, especially in the first years of distance learning have significantly poorer success rate than full-time students. It is necessary to improve the quality of studies, offer quality learning materials and ensure good governance study. To a considerable degree of improvement contributes use e-conference rate and increase students' activity during the semester. More distance students do not complete the courses. However, the average rating of successful distance and full-time students is the same. Therefore, it is essential for quality distance learning to ensure quality distance learning and regular activities of distance students. Our experience suggests that high-quality online tutorials should include solved examples, video tutorials, video lectures, self-tests and an appropriate amount of homework. The most significant problem lies in abilities of students to manage their studies. Part of students is unable to plan the well-proportioned schedule for learning. They often start to learn at the end of the semester before course exam. They are requested to finish several courses successfully and that why they don't have enough time to complete all course requirements, and often they fail in the whole study. Nearly 60% of the distance students do not finish the first year. Full-time students are more successful than distance students. 66% of full-time students completing the courses. Differences between the groups are small in subsequent semesters.

References

1. Bozkurt, A., Akgun-Ozbek, E., Yilmazel, S.: Trends in distance education research: a content analysis of journals 2009–2013. Int. Rev. Res. Open Distrib. Learn. 16(1) (2015). ISSN 1492-3831
2. Cavanaugh, C., Gillan, K.: The effects of distance education on K-12 students outcomes: a meta-analysis. http://files.eric.ed.gov/fulltext/ED489533.pdf. Accessed 2004
3. Chatura, A., Laziuk, B., Galazkowski, R., Wejnarski, A., Roszak, M., Kolodziejczak, B.: Multimedia and interactivity in distance learning of resuscitation guidelines: a randomised controlled trial. Interact. Learn. Environ. **26**(2), 151–162 (2018). ISSN 1744-5191
4. Chráska, M.: Metody pedagogického výzkumu. Základy kvantitativního výzkumu. Grada Publishing (2007). ISBN 978-80-247-1369-4

5. Englund, C., Olofsson, A.: D., Price, L.: Teaching with technology in higher education: understanding conceptual change and development in practice. High. Educ. Res. Dev. **36**(1), 73–87 (2016). ISSN 1469-8366
6. Hannay, M., Newvine, T.: Preception of distance learning: a comparison of online and traditional learning. J. Online Learn. Teach. ISSN 1558-9528. http://jolt.merlot.org/05011.htm. Accessed 2005
7. Levy, Y.: Comparing dropouts and persistence in e-learning courses. Comput. Educ. **48**(2). ISSN 0360-1315 (2007)
8. Klimová Frydrychová, B.: Teaching and learning enhanced by information and communication technologies. Procedia Soci. Behav. Sci. **186** (2015). ISSN 1877-0428
9. Markova, T., Glazkova, I., Zaborova, E.: Quality issues of online distance learning. Procedia Soc. Behav. Sci. **237**, 685–691 (2017). ISSN 1877-0428
10. McPhee, I., Marks, D., Duffy, I.: Comparison of equated learning for online and on-campus postgraduate students on academic achievement. Univ. Fraser Valley Res. Rev. **4**(2) (2012)
11. Okaz, A.: Integrating blended learning in higher education. Procedia Soc. Behav. Sci. **186** (2015). ISSN 1877-0428
12. Ozturk, D., Eyikara, E., Baykara, G.Z.: The opinions of nursing students regarding the first implementation of distance education. World J. Educ. Technol. Curr. Issues **9**(2), 51–58 (2016)
13. Paechter, M., Maier, B.: Online or face to face? Students' experiences and preferences in e-learning. Internet Higher Educ. **2** (2014). ISSN 1096-7516
14. Porter, W.W., Graham, C.R., Bodily, R.G., Sandberg, S.D.: A qualitative analysis of institutional drivers and barriers to blended learning adoption in higher education. Internet High. Educ. **28**, 17–27 (2016). ISSN 1096-7516
15. Řehák, J., Brom, O.: SPSS, Praktická analýza dat. Computer Press (2015). ISBN 978-80-251-4609-5
16. Shachar, M., Neumann, Y.: Differences between traditional and distance education academic performances: a meta-analytic approach. Int. Rev. Res. Open Distance Learn. **4**(2) (2003). ISSN 1492-3831
17. Van de Vord, R.: Distance students and online research: promoting information literacy through media literacy. Internet High. Educ. **13**(3) (2010). ISSN 1096-7516
18. Zhang, L., Worthington, A.C.: Scale and scope economies of distance education in Australian universities. Stud. High. Educ. **42**(9), 1785–1799 (2017). ISSN 0307-5079

Information Technology Applications: Special Session on Language Technologies

Word and Phrase Dictionaries Generated with Multiple Translation Paths

Jouko Vankka(✉), Christoffer Aminoff, Dmitriy Haralson, and Janne Siipola

Department of Military Technology, National Defence University, Helsinki, Finland
jouko.vankka@mil.fi

Abstract. Methods used to learn bilingual word embedding mappings, which project the source-language embeddings into the target embedding space, are compared in this paper. Orthogonal transformations, which are robust to noise, can learn to translate between word pairs they have never seen during training (zero-shot translation). Using multiple translation paths, e.g. Finnish → English → Russian and Finnish → French → Russian, at the same time and combining the results was found to improve the results of this process. Four new methods are presented for the calculation of either the single most similar or the five most similar words, based on the results of multiple translation paths. Of these, the *Summation* method was found to improve the P@1 translation precision by 1.6% points compared to the best result obtained with a direct translation (Fi → Ru). The probability margin is presented as a confidence score. With similar coverages, the probability margin was found to outperform probability as a confidence score in terms of P@1 and P@5.

Keywords: Natural language processing · Distributed word vectors word2vec · Monolingual mappings · Multipath translations Translation matrix

1 Introduction

Parallel corpora, where the same content appears in two languages, as well as bilingual mappings of words and phrases generated through such corpora, commonly form the basis for statistical machine translation. As a consequence of the large amount of resources that is needed in order to create these with high quality, the number of language pairs for which they are available is scarce [9]. Finnish and Russian are one of many language pairs for which there are no readily available parallel corpora.

Monolingual mappings represent words and phrases in a high-dimensional vector space, and these monolingual embeddings have been shown to encode syntactic and semantic relationships between elements of language. These methods often construct word vectors through a context matrix, then either use the vectors directly, perform some factorisation of the matrix, or alternatively use the

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 505–520, 2018.
https://doi.org/10.1007/978-3-319-99972-2_42

context in a neural network that produces vectors for each word. The information captured by these embeddings can be exploited for bilingual translation by learning a translation matrix that allows one to match relative positions across two monolingual vector spaces [3,4,6–9,11,12,15,18,19,22,24]. The assumption that a word appears in similar contexts and is distributed within its language similarly to its equivalent in another language is key to translation through a learned matrix.

In this paper, a set of simple methods is introduced with which one can translate between languages using multipath translation, making use of and combining information gained in indirect and direct translation, or in multiple indirect translation paths, in order to improve translation accuracy. These methods can also combine translation results from multiple noisy translation paths (e.g. Fi → En → Ru and Fi → Fr → Ru) in order to obtain a lower translation error probability.

Figure 1 illustrates the projection of Finnish word vectors into the Russian (Ru) vector space using translation matrices. The Finnish (Fi) vectors were translated directly (Fi → Ru, blue labels), as well as indirectly (Fi → En → Ru, green labels) into the Russian-language space. Then, both translated vectors, along with their respective nearest Russian neighbour vectors (these two could coincide), were projected into a two-dimensional space using t-distributed stochastic neighbour embedding (T-SNE) [14] and displayed with their labels. Figure 1 illustrates the close alignment of Finnish-language words with their Russian translations (red labels). Even when a projection returned an "incorrect" translation (as per the bilingual dictionary, orange labels), the translation was either a plausible alternative, e.g. 'hävitys' → 'разрушение' (in English, 'destruction'), or a related word, as in 'kuiskata' → тихо ('whisper' → 'quietly').

2 Related Work

Traditionally, the development of practical applications for machine translation has been centred on individual language pairs [19]. When one wishes to translate between languages for which little to no parallel data and few dictionaries are available, the most straightforward approach is indirect translation, also known as explicit bridging. Here, one translates to an intermediate language first and then to the desired target language [3,10,21]. As a lingua franca, the English language has large amounts of readily available data in it, and is thus one of the most common intermediate languages. Indirect translation does, however, suffer from two major shortcomings: it requires substantially more time than direct translation and information may be lost while translating through the intermediate language, leading to reduced accuracy [10]. To the best of the authors' knowledge, the algorithms presented are novel ways to combine translation results from multiple translation paths (e.g. Fi → Ru and Fi → En → Ru) in order to obtain a lower translation error probability.

Mikolov et al. [15] establish the distance between the computed vector and the closest word vector as a plausible confidence measure. In this paper, the probability margin is introduced as an alternative. The aim of using the probability

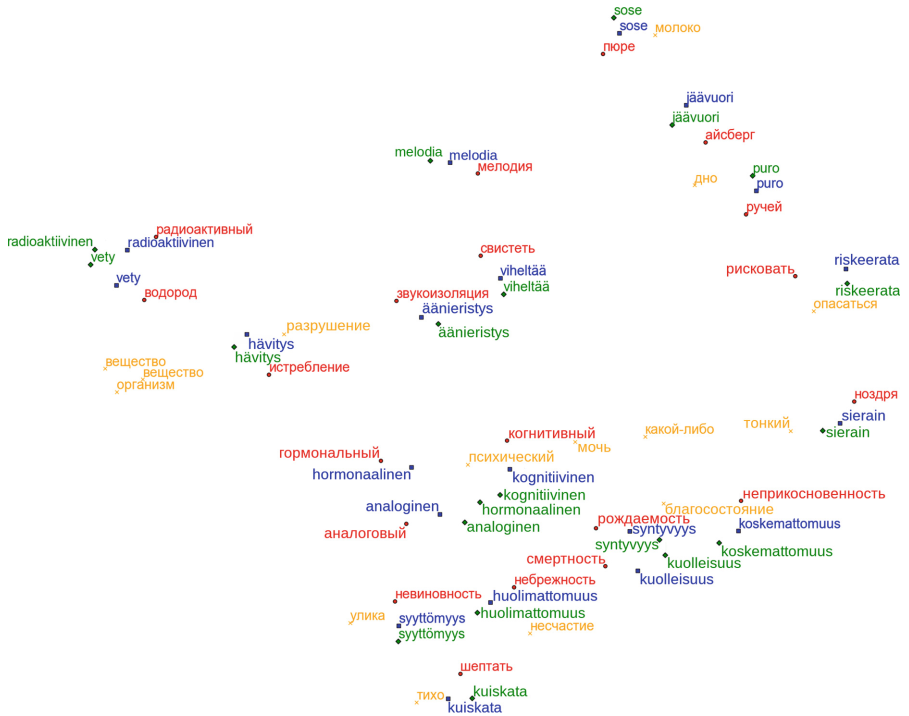


Fig. 1. T-SNE plot of 20 Finnish words projected through two different paths into the Russian embedding space, and their translations.

margin is to filter out more "close races" between high-probability translation candidates in favour of translations where the first-ranked word has a clear lead despite having a low probability. The softmax function can be used to estimate confidence (probability) in a translation prediction made by a translation matrix. If one applies the translation matrix to a word vector in the source word space and obtains a vector in the target space then, if the translation has a low probability or probability margin, it is likely to be inaccurate.

3 Corpus and Word Embedding Vectors

The Finnish-language models were built using word2vec architecture [16]. The models were taught on approximately 2.6 billion tokens from the Suomi24 corpus [1] of Finnish-language discussions on a popular web forum. A lemmatisation pipeline was used in order to produce morphological analyses and lemmas for the Suomi24 corpus [1] of Finnish-language discussions on a popular web forum [23]. The Finnish word2vec CBOW models were built using dimension size 300, window size 15, a minimum frequency for words set at 50, and with hierarchical softmax enabled and negative sampling disabled. The word and phrase translation used online monolingual corpora and word2vec [16]/fastText [5] models in the Russian, French, Swedish and English languages as shown in Tables 1 and 2.

Table 1. Corpus statistics

Language	Corpus	Corpus size (words)	Ngrams
Fi	Suomi24 [1]	2.6 billion	2
Ru	Russian National Corpus + Russian Wikipedia [20]	600 Million	2
Fr	French Wikipedia [13]	1 billion	1
Se	Swedish Wikipedia [2]	86 Million	2
En	GoogleNews [17]	100 billion	2

Table 2. Monolingual word2vec/fastText models

Language	Vocabulary Size	Embedding size	Method
Fi	411 215	300	CBOW
Ru [20]	392 339	300	CBOW
Fr [13]	50130	300	CBOW
Se [2]	1 143 000	300	Skip-gram
En [17]	3 000 000	300	CBOW

4 Translation Matrix

The training dictionaries, containing paired words that were translations of each other, were obtained by scraping Wiktionary and through Google Translate. The training dictionaries were then used to train a translation matrix, a linear mapping from the source space to the target space. The following least-squares error minimisation problem served as the training objective [15].

$$\min_{W} \|Y - WX\|^2 \tag{1}$$

This returns the translation matrix W that minimises (1), where X and Y are word vector matrices for the source and target space of the training data, respectively. After training, the translation of a source word x_i to the target language is done by finding the target word y_j most similar to $x_i W$.

In articles [4, 22, 25], it is argued that the translation matrix should be orthogonal. The one attained in (1) isn't. In order to learn the orthogonal translation matrix, the following maximisation objective is considered

$$\max_{O} \sum_{i=1}^{n} y_i^{\intercal} O x_i, \text{subject to } O^{\intercal} O = I \tag{2}$$

where O denotes the orthogonal translation matrix and I is an identity matrix. The solution to this maximisation problem is the singular value decomposition of $Y^\intercal X$ [22]. The matrix $S = YOX^\intercal$ computes the similarities between all possible pairs of source and target words under the orthogonal transformation O. X and Y are word vector matrices for each language in which each row contains a single word vector denoted by lowercase x and y. The matrix element

$$S_{ij} = y_j^\intercal (Ox_i) \tag{3}$$

evaluates the similarity between the jth source word and the ith target word [22].

[6] identified the presence of hubs (hubness) as a major flaw in nearest neighbour retrieval. A word is referred to as a hub when it appears as the nearest target word to a large number of different source words, which leads to decreased translation accuracy. Softmax is a function that can be used to estimate confidence in a translation prediction made by a similarity matrix. The equation for the softmax is

$$P_{j \to i} = \frac{e^{\beta S_{ij}}}{\sum_m e^{\beta S_{mj}}}. \tag{4}$$

According to [6], the hubness problem is mitigated by inverting the softmax, and normalising the probability over source words rather than target words [22]

$$P_{j \to i} = \frac{e^{\beta S_{ij}}}{\sum_m e^{\beta S_{im}}}. \tag{5}$$

In (5) β should have a value that maximises

$$\max_\beta \sum_{\text{pairs } ij} \ln(P_{j \to i}) \tag{6}$$

and is called the inverse temperature. The numerical method was used to solve the inverse temperature in each test.

When using the inverted softmax in translation, the target word that maximises this probability is selected. The denominator in Eq. (5) grows with the number of source words for which the ith target word is the nearest, making it large for hubs. This in turn significantly reduces a hub's chances of being selected, thereby reducing hubness.

5 Multipath Translation

In indirect translation, the source word is translated to the target language via an intermediate language. To do this, one trains a matrix $M_{s,i}$ to transform word vectors from the source space to the intermediate space. Similarly, one trains a matrix $M_{i,t}$ to transform word vectors from the intermediate space to the target space. One obtains a transformation M from the source space to the target space by multiplying these two matrices together,

$$M = M_{i,t} M_{s,i}. \tag{7}$$

If the two translation matrices are orthogonal, then obviously the resulting translation matrix M is also orthogonal.

Now, by changing the intermediate language, one can have several paths through which one can translate. A set of methods combining information from several paths is presented below.

Summation is a method where one sums the softmax/inverted softmax values of every path. So for a source word x_s one calculates the value of the inverted softmax for a target word y_t and does so for every path. Summing the values gives

$$P_{s \to t} = \sum_i P_{s \to i \to t} \tag{8}$$

where the summation index i ranges over all paths. $P_{s \to i \to t}$ is the (inverted) softmax with translation matrix $M = M_{i,t} M_{s,i}$ defined in (7), or $M = M_{s,t}$ when i points to a path without an intermediate. The target word with the highest $P_{s \to t}$ value is chosen for translation.

The idea behind the *Multiplication* method is similar to Summation, with the summation of probabilities simply being replaced by multiplication. The resulting value $P_{s \to t}$ is still a probability. Formally, this can be expressed as follows:

$$P_{s \to t} = \prod_i P_{s \to i \to t} \tag{9}$$

where multiplication index i runs over all paths in a manner similar to (8).

The third method shall be called the *Margin* method. Here, in each path, one ranks the target words according to their (inverted) softmax value. Then, one compares the paths by taking the difference of the (inverted) softmax values of the first- and second-placed target word in each. Finally, this difference is multiplied by the (inverted) softmax value of the highest-ranked target word. This way one obtains a "margin" value for the rank 1 target word in each path, where both its score and the magnitude of its lead over alternatives are taken into account. The first-place target word of the path with the highest margin value is chosen as the translation. In the Margin method, only one word is chosen, and therefore cannot be used for the calculation of neighbourhoods containing more than one word.

In the fourth method, which shall be called *Combomax*, one ranks all the inverted softmax values from all of the paths into a single list. Each target word is assigned the highest value given to it by any individual path. The target word with the highest resulting value, i.e. rank one, is chosen as the translation. Regarding neighbourhoods, it is good to note that, if one has, for example, 3 paths, then one target word will always have 3 positions in the ranking list. However, only unique words are chosen and ranked. Thus, for example, if one target word takes the first 3 positions in the ranking list, it is still only included once in the neighbourhood of the five most similar words.

6 Results with Datasets

The word pairs used in the test were constructed by translating the words of the source-language corpus into the target language in one of two ways: either through Google Translate in 500-word batches, or by extracting translations for each word from Wiktionary (in some cases also from interlingual links in Wikipedia articles as a last resort), both in order of word frequency. The former method sometimes gave translations of poor quality while the latter failed to find translations for some words, in which case the word was simply disregarded.

Table 3. The translation accuracy of the various language pairs tested. Dictionary entries obtained from Google Translate, except for Fi → Ru and Ru → Fi, where Wiktionary was used.

	Non-orthogonal matrix (W) see Eq. (1)		Orthogonal matrix (O) see Eq. (2)		Non-orthogonal matrix (W) see Eq. (1) and Inverted Softmax see Eq. (5)		Orthogonal matrix (O) see Eq. (2) and Inverted Softmax see Eq. (5)	
	P@1	P@5	P@1	P@5	P@1	P@5	P@1	P@5
Fi → Ru	30.8%	56.1%	28.2%	47.0%	35.5%	59.0%	37.4%	61.1%
Ru → Fi	31.0%	55.1%	37.2%	61.5%	31.0%	52.4%	38.9%	60.9%
En → Ru	49.1%	65.9%	51.0%	68.0%	47.8%	66.8%	49.8%	67.7%
Ru → En	45.0%	58.9%	40.9%	57.0%	42.4%	56.7%	42.4%	59.1%
Fi → En	36.8%	50.5%	34.4%	50.8%	41.4%	59.1%	37.1%	54.5%
En → Fi	43.5%	62.5%	44.6%	63.7%	43.0%	62.8%	43.3%	62.1%
Fi → Fr	18.2%	36.8%	22.7%	41.3%	17.4%	33.5%	24.0%	41.2%
Fr → Fi	13.8%	31.9%	15.3%	31.8%	16.9%	36.2%	18.5%	36.8%
Fi → Se	15.8%	32.2%	25.3%	35.0%	27.2%	38.1%	27.2%	37.5%
Se → Fi	31.8%	49.1%	28.4%	46.0%	34.4%	50.2%	32.1%	48.0%

Table 4. Translations for the Finnish word 'kärry', meaning 'cart' (or colloquially 'car') within P@5. A word in bold is a match with its corresponding dictionary entry. The Swedish translations did not have a hit (dictionary entry: 'kärra'), but were semantically similar.

kärry			
Ru	Se	En	Fr
тележка	släpvagn	truck	**chariot**
телега	lastbilen	gooseneck trailer	remorque
тачка	lastar	tractor	camion
грузовик	lastbil	car	wagon
машина	släpvagnen	**cart**	charrette

Table 5. The accuracy of indirect Finnish-to-Russian translations, where the intermediate languages were English and French. Dictionary entries scraped from Wiktionary.

	Non-orthogonal matrix (W) see Eq. (1)		Orthogonal matrix (O) see Eq. (2)		Non-orthogonal matrix (W) see Eq. (1) and Inverted Softmax see Eq. (5)		Orthogonal matrix (O) see Eq. (2) and Inverted Softmax see Eq. (5)	
Path	P@1	P@5	P@1	P@5	P@1	P@5	P@1	P@5
Fi → En → Ru	16.0%	33.2%	30.1%	50.9%	24.0%	44.2%	30.7%	50.9%
Ru → En → Fi	6.8%	24.7%	26.0%	45.2%	14.7%	31.9%	26.5%	45.0%
Fi → Fr → Ru	13.0%	31.0%	27.6%	48.2%	21.1%	41.0%	28.1%	47.7%
Ru → Fr → Fi	12.3%	30.3%	24.0%	45.1%	17.0%	33.1%	24.2%	45.5%

Translation matrices were then trained with the first 5000 word pairs where both the source and target word had an embedding in their respective spaces, i.e. pairs that didn't were replaced by pairs beyond the 5000th. Test translations were then obtained for the source word in each word pair by transforming the source vector space with the appropriate matrix, and calculating the (inverted) softmax value for each word in the target space. In single-path translations, the target word(s) with the highest such value was (were) chosen as the translation(s). The quality of the obtained translations was measured in terms of accuracy metrics P@1 and P@5 (i.e. the percentage of cases in which the correct translation pair was retrieved as the most similar or among the five most similar words from the other language, respectively). The latter of these allows for a few synonyms that may outrank the dictionary entry.

This paper investigated how translation accuracy is influenced by the similarity measure (cosine similarity vs. inverted softmax), using a non-orthogonal vs. an orthogonal translation matrix, direct vs. indirect translation, multipath translation, as well as the size of the corpus. Softmax and cosine similarity are equivalent when the vectors are normalised. The performance of translation from English to Russian significantly exceeded that from Finnish to Russian as shown in Table 3, which is most likely due to the size of the English corpus shown in Table 1. In articles [4,22,25] it is argued that the translation matrix should be orthogonal. An orthogonal transformation preserves the lengths of vectors and angles between vectors, but reduces degrees of freedom. Table 3 suggests that the optimal retrieval approach would be able to tune continuously between an orthogonal and a non-orthogonal matrix in direct translation between individual language pairs. In general, the inverted softmax (last two columns in Table 3) performed better than cosine similarity (second and third column in Table 3) as a similarity measure.

When one examines Table 4, one can see that some translations that were only found in P@5 were in fact correct. But since for every other language the dictionary entry provided by Google Translate (in bold) had a different

Table 6. Multipath translations and their combinations using the methods described in Sect. 5.

	Fi→En→Ru		Fi→Ru and Fi→En→Ru		Fi→Ru and Fi→En→Ru and Fi→Fr→Ru		Fi→En→Ru and Fi→Fr→Ru		Fi→Fr→Ru		Fi→Ru and Fi→Fr→Ru	
Method	P@1	P@5	P@1	P@5	P@1	P@5	P@1	P@5	P@1	P@5	P@1	P@5
Summation see Eq. (8)	30.9%	50.9%	39.0%	60.0%	36.9%	59.7%	32.1%	53.4%	27.9%	47.9%	36.7%	59.4%
Multiplication see Eq. (9)	30.9%	50.9%	37.5%	58.8%	35.5%	58.6%	31.8%	52.8%	27.9%	47.9%	35.6%	58.5%
Margin see Sect. 5	30.9%	N/A	38.2%	N/A	36.2%	N/A	31.0%	N/A	27.9%	N/A	36.3%	N/A
Combo-max see Sect. 5	30.9%	50.9%	37.4%	60.8%	36.4%	60.2%	30.3%	51.1%	27.9%	47.9%	36.6%	60.2%

translation associated with the word, only the translation to French was marked as correct in P@1. The translated words, however, are semantically similar to the correct translation. For instance, the Russian P@1 translation for 'kärry' is actually the diminutive form of the dictionary entry. Interestingly, many of the incorrect translations in P@1 were somewhat meaningful and were semantically related to the correct translation. The Swedish P@5 results gave words that, similarly to 'kärry', were related to the transportation of goods (such as 'släpvagn', the word for 'trailer'), while the other three languages all had valid translations within P@5. These results show that when building a dictionary, one correct form of the word is not enough to build a proper training set, and the lack of multiple correct words might not give the best picture of the actual results. The P@1 value therefore tends to be highly underestimated, as synonym translations are counted as mistakes, given that only an exact match counts as a successful translation [15]. For this reason, P@5 may be a more appropriate metric for translation accuracy, given that it doesn't automatically count synonym first-rank translations as errors.

Table 5 displays the results of translation through an intermediate language with orthogonal and non-orthogonal translation matrices. Clearly, the orthogonal translation matrix produces much more robust translations than the non-orthogonal matrix. When one compares the direct orthogonal translations in Table 3 to the orthogonal translations via an intermediate language in Table 5, one can't say that one way is clearly better than the other. Here, translation from Finnish to Russian via English outperformed direct translation from Finnish to

Russian, whilst in translation from Russian to Finnish, the direct path performed better.

The results of multipath translations using the methods described in Sect. 5 are shown in Table 6. An orthogonal matrix with inverted softmax was used, since it was found to be the best combination overall. Among the multipath methods, the Summation and Combomax methods (see Sect. 5) gave the best results for P@1/P@5 metrics, respectively. For example, in one case where the translation results of both paths were noisy, Fi → En → Ru and Fi → Fr → Ru, the Summation method improved the P@1 results of single-path translation from 30.9% to 32.1% and P@5 results from 50.9% to 53.4%. The best multipath P@1 result was 39.0%, and it was obtained with the Summation method. That beat the best direct Fi → Ru translation (see Table 3 orthogonal with inv-softmax) by 1.6%-points. This suggests that combining information from several translation paths can give one better translations than just direct translation. One can also see that the choice of paths matters: those multipath combinations that lacked the information provided by the direct path Fi → Ru recorded poorer results. It should also be noted that using multiple paths doesn't guarantee improved results, as evidenced, for instance, by the P@1 result for Combomax with the combination of Fi → En → Ru and Fi → Fr → Ru, and by the P@5 result for Multiplication with Fi → Ru and Fi → Fr → Ru.

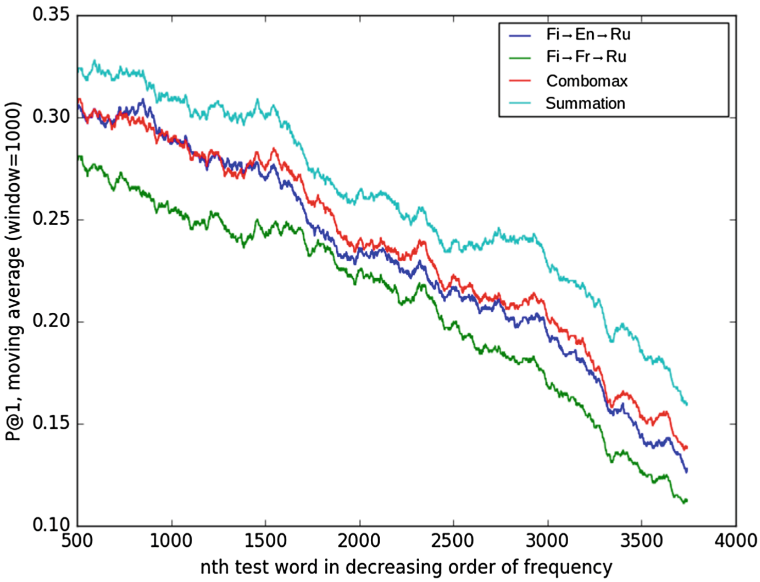


Fig. 2. P@1 translation accuracy for Fi → En → Ru and Fi → Fr → Ru and their combination using methods described in Sect. 5.

6.1 Accuracies of Translation as the Word Frequency Decreases

The accuracies of translation decrease as the frequencies of the source words decrease. In Figs. 2 and 3, the accuracy of the translation is shown for a sliding window of 1,000 words sorted by frequency, starting from rank 5,000 and continuing down to rank 10,000. Each value on the x-axis is the midpoint of a moving average window of translation tests on 1,000 pairs, and the corresponding y-value is the average P@1/5 value of the window. In all cases, the linear transformation was trained on the 5,000 most frequent (available) words and their translations. A decay in accuracy when going into the less common words in the corpora was noted, dropping by 50% at most by the time the 10,000th most frequent word was reached, as can be seen in Figs. 2 and 3. A dropoff in performance was observed for rare words; which could be due to either the difficulty of learning accurate representations for rare words or the lesser similarities between rare words across languages [22].

Six randomly chosen lower-frequency words (with ranks below 10,000) were also selected, the result of translations is shown in Table 7. While some words were translated incorrectly, the semantic similarity was valid for many translations. For instance, the word 'бурундук', the Russian word for 'chipmunk', had its second-ranked translation as 'orava', meaning 'squirrel'. Chipmunks are rodents that behave in a way similar to squirrels. Another interesting set of translations was for 'пузо' , a colloquial word for 'belly'. While there were two hits, the translations tended to refer to the posterior part of the body. All of the translations for 'червяк' (the word for 'worm') and 'изба' ('log hut'/ 'cottage') reflected the word in question. While 'mato' was the exact translation, the other four of them were hyponyms of 'mato', and thus each of them referred to a worm of some kind.

Table 7. P@5 Russian-Finnish translations for words. Correct translations are in bold and words semantically close to the correct translation (mostly hypo- or hypernyms) in italic.

Word	Results				
червяк	*kastemato*	**mato**	*liero*	*onkimato*	*onkiliero*
замутить	virrata	huuhtoa	*vesittää*	loiskia	**pimittää**
пузо	peppu	takamus	*vatsa*	ahteri	**möhömaha**
бурундук	jänis	*orava*	fasaani	rusakko	pihapiiri
сельдь	*kala*	turska	lohi	elintarvike	syötävä
изба	*talo*	**mökki**	*tönö*	*hökkeli*	*röttelö*

6.2 Using Probability or the Probability Margin as a Confidence Measure

Sometimes it is useful to have higher accuracy at the expense of coverage. The probability obtained from the (inverted) softmax (5) or (4) is used as the confidence measure. If the (inverted) softmax values between translation and the

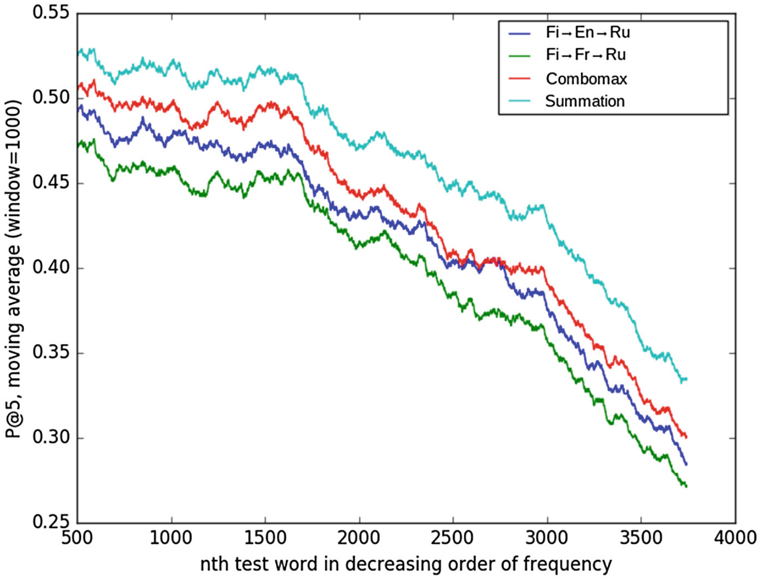


Fig. 3. P@5 translation accuracy for Fi → En → Ru and Fi → Fr → Ru and their combination using methods described in Sect. 5.

most similar target words are low, then the translation is likely to be inaccurate. More formally, the confidence score is defined as

$$\max_{i,k} P_{j \to i}(k) \tag{10}$$

where the maximum is taken over all target words (i) and paths (k). $P_{j \to i}(k)$ is the similarity between the translation of the source word j and target word i in the path k. If this confidence score fails to be higher than the threshold, the translation is disregarded.

A new way to measure confidence is also introduced. One can compare the inverted softmax values of target words that are the most similar and second most similar to the translated source words. If the inverted softmax value of the rank one target word is greatly higher than that of rank two, then the confidence of the translation is considered to be greater than if this difference is small. Formally,

$$\max_{k} \left| P_{j \to i_1}(k) - P_{j \to i_2}(k) \right| \tag{11}$$

where i_1 is the rank one target word and i_2 is the rank two target word in the path k, when target words are ranked according to their inverted softmax values with respect to the translation of the source word j. Again, if this confidence score is lower than the threshold, then the translation is disregarded. Comparing P@1 and P@5 translation accuracy in Tables 8 and 9 shows that with similar coverage, the probability margin outperformed probability as a confidence score overall.

Table 8. Different probability thresholds as confidence measure.

Translation	Probability threshold	Coverage	P@1	P@5
Fi → Ru	0	96.9%	37.4%	61.1%
	0.05	75.5%	42.9%	67.7%
	0.09	48.2%	49.0%	72.2%
	0.2	13.6%	67.6%	85.3%
Fi → En → Ru	0	96.9%	30.9%	50.9%
	0.03	63.5%	38.6%	60.5%
	0.04	48.2%	43.2%	64.9%
	0.06	25.6%	52.3%	75.8%
Fi → Ru and Fi → En → Ru	0	96.9%	37.4%	60.8%
	0.05	75.7%	42.8%	67.4%
	0.09	48.3%	48.9%	72.0%
	0.2	13.6%	67.6%	85.3%
Fi → Ru and Fi → En→Ru and Fi → Fr → Ru	0	97.0%	36.4%	60.2%
	0.05	74.6%	41.6%	67.3%
	0.09	47.6%	47.5%	71.8%
	0.1	43.1%	48.5%	72.2%
Fi → En → Ru and Fi → Fr → Ru	0	97.0%	30.3%	51.1%
	0.03	75.4%	35.8%	58.1%
	0.05	50.3%	40.6%	63.0%
	0.08	22.4%	50.0%	71.0%
Fi → Fr → Ru	0	97.0%	28.4%	48.4%
	0.03	65.0%	35.7%	57.8%
	0.04	51.2%	37.9%	58.6%
	0.06	27.9%	43.0%	65.2%
Fi → Ru and Fi → Fr → Ru	0	97.0%	36.6%	60.2%
	0.05	74.2%	42.5%	67.1%
	0.09	47.3%	48.2%	72.1%
	0.2	11.9%	64.7%	84.9%

Table 9. Different probability margin thresholds as a confidence measure.

Translation	Probability margin threshold $\lvert P_{j \to i}(k)_1$ - $P_{j \to i}(k)_2 \rvert$	Coverage	P@1	P@5
Fi → Ru	0	96.9%	37.4%	61.1%
	0.005	73.2%	45.5%	66.7%
	0.02	43.6%	56.7%	72.9%
	0.04	26.6%	69.2%	81.2%
Fi → En → Ru	0	96.9%	30.9%	50.9%
	0.002	70.7%	38.5%	57.6%
	0.006	47.0%	48.5%	65.7%
	0.01	34.3%	55.4%	71.4%
Fi → Ru and Fi → En → Ru	0	96.9%	37.4%	60.8%
	0.005	73.3%	45.6%	66.7%
	0.02	43.9%	56.3%	72.7%
	0.04	26.4%	69.3%	81.1%
Fi → Ru and Fi → En → Ru and Fi → Fr → Ru	0	97.0%	36.4%	60.2%
	0.005	72.5%	45.0%	68.0%
	0.02	42.6%	55.6%	71.8%
	0.04	26.0%	66.9%	80.8%
Fi → En → Ru and Fi → Fr → Ru	0	97.0%	30.3%	51.1%
	0.003	70.2%	37.5%	57.1%
	0.007	50.0%	47.0%	63.4%
	0.02	22.5%	60.9%	75.6%
Fi → Fr → Ru	0	97.0%	28.4%	48.4%
	0.002	72.4%	34.0%	54.1%
	0.006	46.6%	44.0%	61.6%
	0.01	32.6%	52.5%	66.6%
Fi → Ru and Fi → Fr → Ru	0	97.0%	36.6%	60.2%
	0.005	71.9%	44.9%	67.9%
	0.02	42.0%	56.9%	72.9%
	0.04	25.0%	69.2%	81.2%

7 Conclusion

This paper explored the feasibility of using monolingual language models in order to perform machine translation. This was done using word2vec models of languages that are not closely related (such as Finnish and Russian). The goal was to obtain a lower error probability and a higher reliability by using data from multiple translation paths.

We observed that the multipath method Summation was able to improve translation results compared to a single direct translation path. We also observed that orthogonal translation matrices were robust when the translation path contained an intermediate language, whereas non-orthogonal ones were not.

The improvements in translation accuracy did however come at the cost of efficiency, since the implementation of multipath translation necessitates more complexity and thereby increased translation times. Whether an improvement was achieved or not depended on the specific combination of paths, as well as the multipath method.

References

1. Aller Media Oy: The Suomi 24 Corpus (2014). http://urn.fi/urn:nbn:fi:lb-201412171. 14 May 2015 Version
2. Amazon S3: Swedish Wikipedia (2017). https://s3-us-west-1.amazonaws.com/fasttext-vectors/wiki.sv.vec
3. Ammar, W., Mulcaire, G., Tsvetkov, Y., Lample, G., Dyer, C., Smith, N.A.: Massively multilingual word embeddings. CoRR abs/1602.01925 (2016). http://arxiv.org/abs/1602.01925
4. Artetxe, M., Labaka, G., Agirre, E.: Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 2289–2294 (2016)
5. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. arXiv preprint arXiv:1607.04606 (2016)
6. Dinu, G., Baroni, M.: Improving zero-shot learning by mitigating the hubness problem. CoRR **abs/1412.6568** (2014). http://arxiv.org/abs/1412.6568
7. Faruqui, M., Dyer, C.: Improving vector space word representations using multilingual correlation. In: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, pp. 462–471 (2014)
8. Haghighi, A., Liang, P., Berg-Kirkpatrick, T., Klein, D.: Learning bilingual lexicons from monolingual corpora, pp. 771–779 (2008)
9. Jansen, S.: Word and phrase translation with word2vec. CoRR abs/1705.03127 (2017). http://arxiv.org/abs/1705.03127
10. Johnson, M., et al.: Google's multilingual neural machine translation system: enabling zero-shot translation. CoRR **abs/1611.04558** (2016). http://arxiv.org/abs/1611.04558
11. Koehn, P., Knight, K.: Estimating word translation probabilities from unrelated monolingual corpora using the em algorithm. In: Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence, pp. 711–715. AAAI Press (2000). http://dl.acm.org/citation.cfm?id=647288.721610
12. Koehn, P., Knight, K.: Learning a translation lexicon from monolingual corpora. In: Proceedings of the ACL-02 Workshop on Unsupervised Lexical Acquisition, vol. 9, pp. 9–16. Association for Computational Linguistics (2002)
13. Kyubyong/wordvectors: French Wikipedia (2017). https://github.com/Kyubyong/wordvectors

14. Maaten, L.V.D., Hinton, G.: Visualizing data using t-SNE. J. Mach. Learn. Res. **9**(Nov), 2579–2605 (2008)
15. Mikolov, T., Le, Q.V., Sutskever, I.: Exploiting similarities among languages for machine translation. CoRR abs/1309.4168 (2013). http://arxiv.org/abs/1309.4168
16. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
17. mmihaltz/word2vec-GoogleNews-vectors: GoogleNews (2017). https://github.com/mmihaltz/word2vec-GoogleNews-vectors
18. Rapp, R.: Automatic identification of word translations from unrelated English and German corpora. In: Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics, pp. 519–526. Association for Computational Linguistics (1999)
19. Ruder, S.: A survey of cross-lingual embedding models. CoRR abs/1706.04902 (2017). http://arxiv.org/abs/1706.04902
20. RusVectōrēs: word embeddings for Russian online: Russian National Corpus + Russian Wikipedia (2017). http://rusvectores.org/static/models/ruwikiruscorpora_upos_cbow_300_20_2017.bin.gz
21. Schwenk, H., Tran, K., Firat, O., Douze, M.: Learning joint multilingual sentence representations with neural machine translation. CoRR abs/1704.04154 (2017). http://arxiv.org/abs/1704.04154
22. Smith, S.L., Turban, D.H.P., Hamblin, S., Hammerla, N.Y.: Offline bilingual word vectors, orthogonal transformations and the inverted softmax. CoRR abs/1702.03859 (2017). http://arxiv.org/abs/1702.03859
23. Venekoski, V.: Semantics in Finnish Distributional Language Models. Master's thesis, University of Helsinki (2016)
24. Vulic, I., Korhonen, A.: On the role of seed lexicons in learning bilingual word embeddings. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (2016). http://aclweb.org/anthology/P/P16/P16-1024.pdf
25. Xing, C., Wang, D., Liu, C., Lin, Y.: Normalized word embedding and orthogonal transform for bilingual word translation. In: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1006–1011 (2015)

Sentiment Analysis of Lithuanian Texts Using Deep Learning Methods

Jurgita Kapočiūtė-Dzikienė[1](✉), Robertas Damaševičius[2], and Marcin Woźniak[3]

[1] Vytautas Magnus University, K. Donelaičio 58, 44248 Kaunas, Lithuania
jurgita.kapociute-dzikiene@vdu.lt

[2] Kaunas University of Technology, K. Donelaičio 73, 44029 Kaunas, Lithuania
robertas.damasevicius@ktu.lt

[3] Silesian University of Technology, Kaszubska 23, 44-101 Gliwice, Poland
marcin.wozniak@polsl.pl

Abstract. We describe experiments in sentiment analysis of the Lithuanian texts using the deep learning methods: Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN). Methods used with pre-trained Lithuanian neural word embeddings are tested with different pre-processing techniques: emoticons restoration, stop words removal, diacritics restoration/elimination. Despite the selected pre-processing technique, CNN was always outperformed by LSTM. Better results (reaching an accuracy of 0.612) were achieved with the undiacritized texts and undiacritized word embeddings. However, these results are still worse if compared to the ones obtained using Support Vector Machines or Naive Bayes Multinomial and with the frequencies of words as features.

Keywords: Positive/negative/neutral sentiments · LSTM and CNN methods
Neural word embeddings · The Lithuanian language

1 Introduction

The era of Internet have changed the ways how people express their attitudes and opinions about topics, events, products/services, interactions, etc. It is mainly done via social networks, blogs, web forums, review websites, and Internet comments. All these texts are sentiment rich therefore very beneficial for companies or individuals willing to respond accordingly. Automatic sentiment analysis methods are commonly used for this purpose. Sentiment analysis has been used for analyzing sentiments of tweets on health topics [35], evaluate teachers performance from the student responses in the education domain [30], identify contextual polarity of drugs, movie and restaurant reviews in the business domain [1, 3, 4], examine public opinion on products and services [29], classify e-learners and their topics of interest in their social networks interactions [32], perform opinion mining in tweets [9, 15], analyze sentiment orientation of microblogs [42], and forecast stock prices from sentiments of news signals in the financial markets domain [31].

All sentiment analysis (or sentiment classification) methods can be grouped into two main categories: *dictionary-based* and *machine learning*. The dictionary-based

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 521–532, 2018.
https://doi.org/10.1007/978-3-319-99972-2_43

methods (e.g., [40, 41], etc.) typically rely on the dictionary of sentiment words and syntactic knowledge. An alternative for the dictionary-based methods is machine learning (e.g., [26, 28], etc.). The most popular examples of the standard classification approaches are Support Vector Machines (SVM), Naive Bayes Multinomial (NBM), and k-Nearest Neighbors (kNN). These methods are typically applied on (weighted) frequency-based features of the different feature types (lexical, character, morphological, etc.).

In the recent years, the deep learning methods have gained a lot of interest in many text classification tasks, including sentiment analysis. Therefore our focus in this paper is on this group of methods. As claimed in [25], a Deep Neural Network (DNN) architecture that jointly uses character-, word-, and sentence-level representations achieves the state-of-the-art performance for binary (positive and negative) classification on the Stanford Sentiment Treebank of movie reviews [27] and Stanford Twitter sentiment corpus [11]. Katoh and Ninomiya in [18] solves binary sentiment classification task on the English and Japanese datasets. The authors prove that multi-layer Neural Networks (NNs) work well only on the large-scale datasets, but on the smaller datasets NNs without hidden layers are more suitable. The Convolutional Neural Network (CNN) method was applied on the binary-class Twitter dataset used as the benchmark in the SemEval-2015 competition [36]. The authors have not participated in the competition but their later results could be ranked in the first two positions in both the phrase-level (among 11 teams) and message-level (among 40 teams) subtasks. The experiments described in [12] are performed on the English binary review data obtained from Amazon. Authors used CNN method with Google News *word2vec* embeddings and demonstrated superiority over the baseline methods. A hybrid approach applied on the Hindi language and the dataset of four classes (positive, negative, neutral and conflict) is based on the CNN architecture augmented with to a set of optimized features selected through a multi-objective optimization framework. The augmented optimization vector is afterwards used for training the SVM method [2]. The system, offered by Stojanovski et al. in [39], uses both CNN and Recurrent Neural Network (RNN) to obtain more diverse representations on top of GloVe (Global Vectors) word embeddings. For texts of the variable length both neural networks allow to extract fixed length representation vectors and to use their concatenation. The system, applied on the binary or five classes (positive, negative, neutral, very positive, very negative) benchmark Twitter datasets, was ranked the second best in SemEval-2016 task [39]. The authors in [21] compared two deep learning methods (in particular, RNN and CNN) with neural embeddings and one standard machine learning SVM of unigram, bigram, unigram + bigram features. They experimentally claimed that deep learning methods underperform most of the SVM models on English and Japanese sentiment datasets with either two or three classes. The very interesting description in [7] summarizes sentiment analysis works presented by 32 participants in SemEval-2017 Task 5. The participants experimented with the Financial Microblogs and News data. However, the standard machine learning based technique (rather than deep learning) was ranked the first: it employed linguistic, sentiment lexicon, domain-specific features and Google word embeddings to construct models of ensemble regression algorithms [14]. The second best technique already used the deep learning, in particular, an ensemble-based model that combined CNN with the Long Short-Term

Memory (LSTM) and word embeddings [10]. The competition of in SemEval-2017 Task 4 described in [34] attracted 48 teams. Sentiment analysis was done on the Twitter data of two or five classes. The best-ranked teams used deep learning: e.g., *BB_twtr* employed an ensemble of LSTM and CNN methods with multiple convolution operations [6]; and *DataStories* applied LSTM network with attention mechanism [5].

Although the results of sentiment analysis are controversial (i.e., there is no consensus which method is the best), deep learning has become a dominant paradigm recently.

So far deep learning methods have never been applied on the Lithuanian language for the sentiment classification task and, consequently, this research is the first attempt. Here we deal with the Internet comments dataset of three classes (positive, negative, and neutral) and: (1) compare two deep learning approaches, in particular, LSTM and CNN with neural word embeddings as feature representations; (2) experiment with the different pre-processing techniques (emoticons replacement, stop words removal, diacritics elimination/restoration); and (3) compare the results obtained with the standard machine learning approaches, in particular, SVM and NBM with frequencies of words as features.

2 Methodology

2.1 Formal Definition of the Task

The sentiment classification task in our research can be interpreted as the supervised text classification task.

Let $d_1, d_2, \ldots, d_n$ be the texts (Internet comments) attached to only one of $c_1, c_2, \ldots, c_m$ class labels (where $m = 3$ in our research determines *positive*, *negative* and *neutral* classes). Thus we have a single-label but multi-class $(m > 2)$ classification problem. The used dataset is described in Sect. 2.2.

Let function γ denote a mapping of texts to their appropriate classes: γ: $D \rightarrow C$. A goal is to use a method which could find the approximation of γ. For this reason we test the deep learning approaches described in Sect. 2.3.

2.2 Dataset

In our sentiment classification experiments we have used the dataset of the Lithuanian Internet comments (the same as in [17]). This dataset is of the general domain: it contains opinions about articles on the various topics (politics, sport, health, economics, etc.) presented in the news portal https://www.lrytas.lt/. The Internet comments represent the non-normative Lithuanian language: they are full of out-of-vocabulary words, spelling errors, abbreviations, emoticons, etc. The texts are very short, i.e., ~8.58 words per text.

Texts in this dataset are grouped in three categories: positive, negative, and neutral (for the statistics see Table 1).

Table 1. The statistics about the dataset used in our sentiment analysis tasks

Class label	Numb. of texts	Numb. of tokens	Tokens with emoticons	Tokens without stop words
Positive	1,500	10,455	10,664	8,982
		3,177	4,027	3,941
Negative	1,500	15,000	15,107	11,945
		6,475	7,811	7,716
Neutral	1,500	13,165	13,226	10,427
		5,134	6,391	6,276
Total	4,500	38,621	38,997	31,354
		11,669	14,966	14,923

During our sentiment analysis experiments we have explored the impact of the following pre-processing techniques:

- *No pre-processing*. The texts were lowercased, numbers and punctuation were eliminated.
- *With emoticons*. As proved in [33], the emoticon replacement leads to the higher classification accuracy. In our experiments we have used 32 groups of emoticons, where each group was related with the appropriate sentiment word (presented in its main vocabulary form). For example, all emoticons of this group ":(", ":-("', ":-c", ":-[", etc. are related with the word "liūdnas" (sad). When using this pre-processing technique, the words were lowercased; the detected emoticons were replaced with the appropriate sentiment words; numbers and punctuation were eliminated.
- *No stop words*. Intuitively, words as acronyms, conjunctions or pronouns are too often and too short to carry important sentiment information. However, Spences and Uchyigit in [38] claim that interjections are strong indicators of subjectivity. Here we have used a list of 555 stop words and that list does not contain interjections. During pre-processing all words were lowercased, stop words, numbers and punctuation were eliminated.
- *Diacritics elimination*. The Lithuanian language uses the Latin alphabet supplemented with the diacritics: *ą*, *č*, *ę*, *ė*, *š*, *ų*, *ū*, *ž*. However, diacritics in the non-normative Lithuanian texts are sometimes omitted, therefore the same word can be found in both cases, i.e., with or without diacritics. It increases the data sparseness, which, in turn, negatively affects the accuracy. One of the solutions is to distort the data by replacing diacritized letters with their ASCII equivalents. During this pre-processing the words were undiacritized, lowercased, numbers and punctuation were eliminated.
- *Diacritics restoration* is another direction which could decrease the data sparseness and increase text quality. As proved in [16], the language modelling method is the best coping with the Lithuanian diacritization problems. We have applied this language modelling method with the bi-gram back-off strategy on our non-normative texts. The unigrams ($\sim$2.3 million) and bigrams ($\sim$58.1 million) were learned from the corpus of $\sim$234 million running tokens.

2.3 Deep Learning Methods

In this research we have tested two popular deep learning methods:

- *Long Short Term Memory* (LSTM) [13] as the type of RNN with the memory unit capable of learning long-term dependencies. The memory unit (a cell) is composed of input, output and forget gates and is responsible for remembering values over arbitrary time intervals. The output of 256 nodes in the LSTM layer is an input to the fully connected softmax layer which output is the probability distribution over classes.
- Convolutional Neural Network (CNN) [20]. The convolution is performed on the concatenated vectors of words (taken from the text sequentially). The filter is applied on the window of these words to produce a new feature. The filter of 3, 4, and 5 widths was used in this approach. The feature map is produced when the filter is applied on each possible window of words in that text. Afterwards, the max-over-pooling operation over the feature map of the particular filter determines a single maximum value. The features from different filters are passed to a fully connected layer, whose output is the probability distribution over classes. The detailed description of this approach can be found in [19].

For testing LSTM and CNN we have used their implementations in *deeplearning4j* [8] – the open-source distributed deep learning library for the Java Virtual Machine. However, the existing implementations could solve only binary classification tasks, therefore necessary adjustments to multi-class classification were performed by the authors of this research. Both methods truncate the texts exceeding 256 words, all other parameters were set to their default values.

Deep learning methods for the sentiment classification were used after plugging in the appropriate Lithuanian neural *word2vec* embeddings. These embeddings were generated with the same *deeplearning4j* software, the continuous bag-of-words (CBOW) plus negative sampling method and 300 dimensions (the detailed explanation of the method is in [23, 24]). Using the corpus of $\sim$234 million running words, it resulted in 687,947 neural word embeddings for the Lithuanian language.

The Lithuanian word embeddings are generated from the normative Lithuanian texts. Intuitively they cannot be effective on the undiacritized texts (especially used with the diacritics elimination pre-processing technique described in Sect. 2.2). Undiacritized non-normative texts are already distorted texts, therefore one of the solutions is also to distort the word embeddings by removing diacritics. Such removal might cause disambiguation problems (because both words with and without diacritics may exist in the Lithuanian language and have absolutely different meanings). E.g., "mama" (mother in nominative) and "mamą" (mother in accusative), but "karštas" (hot), "karstas" (coffin) (has different meanings). The list of embeddings could not contain the same words; therefore we had to eliminate less frequent. It resulted in 632,435 undiacritized word embeddings in total. While such distortion might be harmful for our solving task, however, we still take a risk to see the impact on the overall accuracy.

3 Experimental Set-Up and Results

The experiments were carried out on the dataset and the different pre-processing techniques described in Sect. 2.2 using the deep learning methods described in Sect. 2.3.

The experiments were performed using stratified 10-fold cross validation and evaluated using *macro accuracy* and *f-score* (averaged over classes and folds) performance measures [37].[1]

To determine whether differences between the results are statistically significant we have performed McNemar's [22] test with the selected significance level equal to 95%. It means that the calculated *p-value* must be below 0.05 so that the differences would be considered as statistically significant.

Since the used dataset is balanced, both random $\sum P(c_i)^2$ and majority $\max(P(c_i))$ baselines (where $P(c_i)$ is the probability of the class c_i) are equal to 0.33.

The Fig. 1 summarizes the results obtained on the original Lithuanian sentiment dataset using different pre-processing techniques. For the comparison purposes we also include the SVM and NBM results taken from [17]. However, the difference is not only in the classification method, but also in the features representation. The standard machine learning approaches, in particular, SVM and NBM, use the "traditional approach" where each feature corresponds to some word from a dictionary and represents a number of times that word occurs in the document.

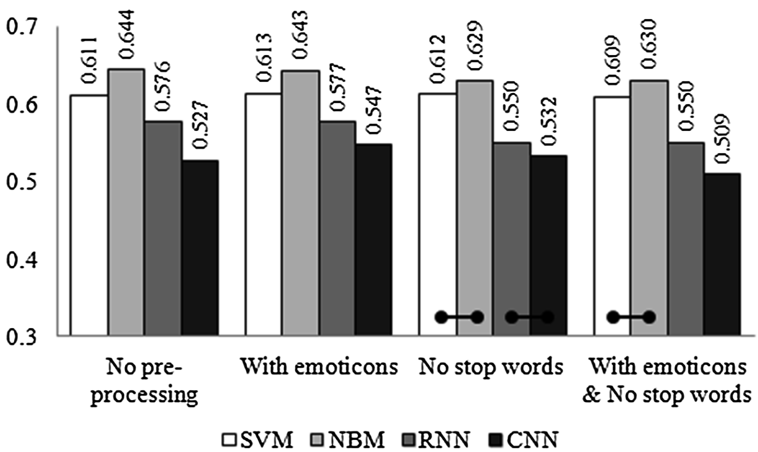


Fig. 1. Accuracies with different pre-processing techniques. Black connecting lines in each group of columns connect results of which differences are not statistically significant.

[1] Since *f-score* values demonstrated the same trend as the *accuracy*, we do not present them.

The Figs. 2 and 3 summarize the results obtained with the eliminated and restored diacritics, respectively. Since we do not have SVM and NBM results with the restored diacritics, they are not presented in Fig. 3.

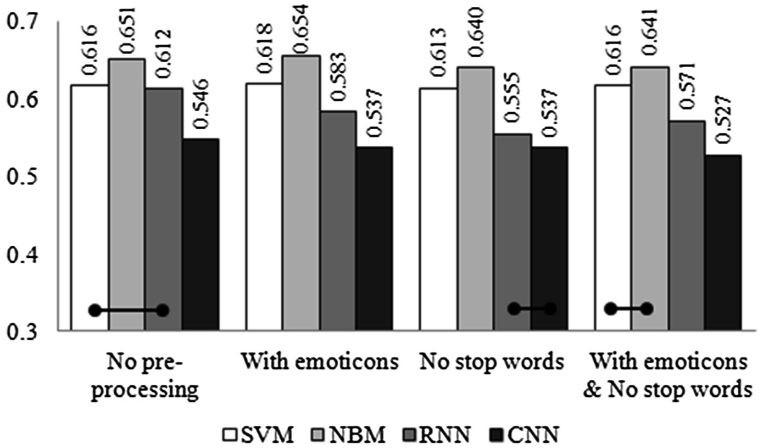


Fig. 2. Accuracies with eliminated diacritics and different pre-processing techniques. Black connecting lines in each group of columns connect results of which differences are not statistically significant.

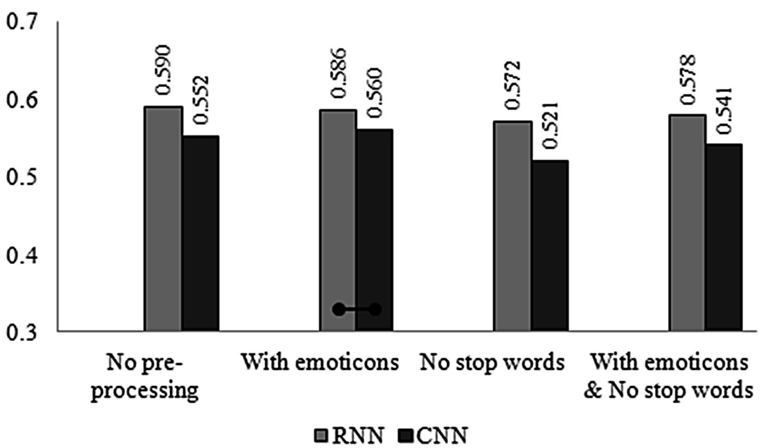


Fig. 3. Accuracies with restored diacritics and different pre-processing techniques. Black connecting lines in each group of columns connect results of which differences are not statistically significant.

4 Discussion

From Figs. 1, 2 and 3 we can note that our sentiment analysis results are reasonable, because they outperform the random and majority baselines (equal to 0.33).

The diacritics restoration and elimination pre-processing techniques in most of the cases improved the accuracy (except for CNN with stop words removal in diacritics restoration and CNN with emoticons restoration in diacritics elimination). Unfortunately the increase was statistically significant only for RNN with no pre-processing + diacritics elimination ($p = 0.008$); for RNN with emoticons & stop words + diacritics restoration ($p = 0.039$); for CNN with emoticons & stop words + diacritics restoration ($p = 0.029$). Calculated p-values are summarized in Table 2.

Table 2. *p-values*, denoting the differences between the accuracies presented in Fig. 1 and accuracies presented in Fig. 2 (of eliminated diacritics) and Fig. 3 (restored diacritics). Underlined *p-values* present statistically significant differences.

	RNN		CNN	
	Eliminated diacritics	Restored diacritics	Eliminated diacritics	Restored diacritics
No pre-processing	<u>0.008</u>	0.320	0.182	0.088
With emoticons	0.661	0.483	0.472	0.352
No stop words	0.742	0.123	0.760	0.420
With emoticons & No stop words	0.135	<u>0.039</u>	0.235	<u>0.029</u>

Despite the differences not always are statistically significant, it allows us to make the statement that diacritics treatment is important when dealing with the non-normative texts. Word embeddings were trained from the normative Lithuanian texts, which means that any undiacritized word (not having undiacritized equivalents) found in the text is treated as out-of-vocabulary and is not considered during sentiment classification. Therefore any solution (restoration or elimination of diacritics) helps finding word equivalents among the neural word embeddings. Despite, marginally the best results (reaching 0.612 of accuracy) were achieved with the undiacritized texts and LSTM, in most of the cases diacritics restoration (used together with the other pre-processing techniques as emoticons restoration or stop words removal) outperformed diacritics elimination. The diacritics restoration process is of course not 100% accurate and even slight differences (mostly due to the different morphological forms) cause words' mapping to absolutely different word embeddings (e.g., "geras" (good in nominative) and "gerą" (good in accusative) are represented by different word vectors). Moreover, word embeddings are generated from the normative Lithuanian texts, which mean that jargon, borrowings, foreign language words expressing sentiments are not included. Summarizing, diacritics restoration and word embeddings generation from the non-normative texts could be possible improvement direction for the future research.

The stop words removal in most of the cases decreased the accuracy (except for the CNN on the original data). It means that even very short words carry important sentiment information. The results of emoticon restoration were mixed.

As we can see from Figs. 1, 2 and 3, CNN is always outperformed by LSTM, and in most of the cases the differences in the results are statistically significant (the exact *p-values* are in Table 3). Therefore we can claim that LSTM is more suitable for our sentiment recognition task. Deep learning methods were outperformed by SVM and in all cases significantly outperformed by NBM. These findings coincide with presented in [21]. However, as claimed in [18], the deep learning approaches work better on larger datasets. Consequently, the experiments on larger datasets are in our future plans.

Table 3. *p-values* denoting the statistical significance between different methods and pre-processing techniques

		RNN&CNN	RNN&SVM	RNN&NBM	CNN&SVM	CNN&NBM	SVM&NBM
Original	No pre-processing	0.001	0.010	0.000	0.000	0.000	0.010
	With emoticons	0.033	0.007	0.000	0.000	0.000	0.020
	No stop words	0.214	0.000	0.000	0.000	0.000	0.192
	With emoticons & No stop words	0.005	0.000	0.000	0.000	0.000	0.106
Eliminated diacritics	No pre-processing	0.000	0.761	0.000	0.000	0.000	0.006
	With emoticons	0.001	0.008	0.000	0.000	0.000	0.005
	No stop words	0.205	0.000	0.000	0.000	0.000	0.036
	With emoticons & No stop words	0.002	0.000	0.000	0.000	0.000	0.052
Restored diacritics	No pre-processing	0.006					
	With emoticons	0.058					
	No stop words	0.000					
	With emoticons & No stop words	0.007					

5 Conclusions

The main contribution of this research – the Lithuanian sentiment analysis, solved with the new group of methods, in particular, deep learning methods on top of the neural word embeddings. Here we have tested two approaches, in particular, Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN), where CNN was significantly outperformed by LSTM in all our experiments.

Out of several tested pre-processing techniques (in particular, emoticons restoration, stop words removal, (un-)diacritization) marginally the best results (reaching an accuracy of ~ 0.612) were achieved with LSTM, undiacritized texts and undiacritized word embeddings. However, the impact of any other selected pre-processing technique was not statistically significant.

The comparative analysis revealed that the deep learning methods with neural word embeddings significantly underperform Support Vector Machines (SVM) and Naive

Bayes Multinomial (NBM) with the frequencies of words from the dictionary as feature representations.

The limitations of our research are as follows. We have experimented with the rather small sentiment dataset, containing only 1,500 texts in each positive, negative and neutral category. However, to get more significant improvements in accuracy, we need larger annotated dataset.

References

1. Agarwal, B., Mittal, N., Bansal, P., Garg, S.: Sentiment analysis using common-sense and context information. Comput. Intell. Neurosci. **2015**, 1–9 (2015)
2. Akhtar, M.S., Kumar, A., Ekbal, A., Bhattacharyya, P.: A hybrid deep learning architecture for sentiment analysis. In: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pp. 482–493 (2016)
3. Asghar, M.Z., Khan, A., Ahmad, Sh., Qasim, M., Khan, I.A.: Lexicon-enhanced sentiment analysis framework using rule-based classification scheme. PLOS ONE **12**(2), e0171649 (2017)
4. Augustyniak, Ł., Szymański, P., Kajdanowicz, T., Tuligłowicz, W.: Comprehensive study on lexicon-based ensemble classification sentiment analysis. Entropy **18**(1), 4 (2015)
5. Baziotis, Ch., Pelekis, N., Doulkeridis, Ch.: DataStories at SemEval-2017 task 4: deep LSTM with attention for message-level and topic-based sentiment analysis. In: Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval 2017 (2017)
6. Cliché, M.: BB_twtr at SemEval-2017 task 4: twitter sentiment analysis with CNNs and LSTMs. In: Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval 2017 (2017)
7. Cortis, K., et al.: SemEval-2017 task 5: fine-grained sentiment analysis on financial microblogs and news. In: Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval-2017 (2017)
8. Deeplearning4j Development Team: Deeplearning4j: Open-source distributed deep learning for the JVM, apache software foundation license 2.0 (2017)
9. Ferrara, E., Yang, Z.: Quantifying the effect of sentiment on information diffusion in social media. PeerJ Comput. Sci. **1**, e26 (2015)
10. Ghosal, D., Bhatnagar, Sh., Akhtar, Md.Sh., Ekbal, A., Bhattacharyya, P.: IITP at SemEval-2017 task 5: an ensemble of deep learning and feature based models for financial sentiment analysis. In: Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval-2017 (2017)
11. Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. Processing, 1–6 (2009)
12. Guan, Z., Chen, L., Zhao, W., Zheng, Y., Tan, Sh., Cai, D.: Weakly-supervised deep learning for customer review sentiment classification. In: International Joint Conference on Artificial Intelligence, pp. 3719–3725 (2016)
13. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
14. Jiang, M., Lan, M., Wu, Y.: ECNU at SemEval-2017 task 5: an ensemble of regression algorithms with effective features for fine-grained sentiment analysis in financial domain. In: Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval-2017 (2017)

15. Kanavos, A., Nodarakis, N., Sioutas, S., Tsakalidis, A., Tsolis, D., Tzimas, G.: Large scale implementations for twitter sentiment classification. Algorithms **10**(1), 33 (2017)
16. Kapočiūtė-Dzikienė, J., Davidsonas, A., Vidugirienė, A.: Character-based machine learning vs. language modeling for diacritics restoration. Inf. Technol. Control **46**(4), 508–520 (2017)
17. Kapočiūtė-Dzikienė, J., Krupavičius, A., Krilavičius, T.: A comparison of approaches for sentiment classification on lithuanian internet comments. In: Proceedings of the 4th Biennial International Workshop on Balto-Slavic Natural Language Processing, pp. 2–11 (2013)
18. Katoh, K. Ninomiya, T.: Deep learning for large-scale sentiment analysis using distributed representations. In SEMAPRO 2015: The 9th International Conference on Advances in Semantic Processing, pp. 92–96 (2015)
19. Kim, Y.: Convolutional neural networks for sentence classification. In: Empirical Methods in Natural Language Processing, EMNLP, pp. 1746–1751 (2014)
20. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**, 2278–2324 (1998)
21. Lu, Y., Sakamoto, K., Shibuki, H., Mori, T.: Are deep learning methods better for twitter sentiment analysis? In: The 7th International Annual Engineering Seminar, InAES (2017)
22. McNemar, Q.: Note on the sampling error of the difference between correlated proportions or percentages. Psychometrika **12**(2), 153–157 (1947)
23. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR, abs/1301.3781 (2013)
24. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems 26: The 27th Annual Conference on Neural Information Processing Systems 2013, pp. 3111–3119 (2013)
25. Nogueira dos Santos, C., Maira, G.: Deep convolutional neural networks for sentiment analysis of short texts. In: Proceedings of the 25th International Conference on Computational Linguistics (COLING), pp. 69–78 (2014)
26. Pak, A., Paroubek, P.: Twitter for sentiment analysis: when language resources are not available. In: Proceedings of Database and Expert Systems Applications, DEXA 2011, pp. 111–115 (2011)
27. Pang, B., Lee, L.: Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In: Proceedings of the Association for Computational Linguistics (ACL), pp. 115–124 (2005)
28. Pang, B., Lee, L., Vaithyanathan, Sh.: Thumbs up? Sentiment classification using machine learning techniques. In: Proceedings of Conference on Empirical Methods in Natural Language Processing, EMNLP, pp. 79–86 (2002)
29. Paredes-Valverde, M.A., Colomo-Palacios, R., Salas-Zárate, M.d.P., Valencia-García, R.: Sentiment analysis in spanish for improvement of products and services: a deep learning approach. Sci. Program. **2017**, 1–6 (2017)
30. Rajput, Q., Haider, S., Ghani, S.: Lexicon-based sentiment analysis of teachers' evaluation. Appl. Comput. Intell. Soft Comput. **2016**, 1–12 (2016)
31. Ranco, G., Bordino, I., Bormetti, G., Caldarelli, G., Lillo, F., Treccani, M.: Coupling news sentiment with web browsing data improves prediction of intra-day price dynamics. PLOS ONE **11**(1), e0146576 (2016)
32. Ravichandran, M., Kulanthaivel, G., Chellatamilan, T.: Intelligent topical sentiment analysis for the classification of E-learners and their topics of interest. Sci. World J. **2015**, 1–8 (2015)
33. Read, J.: Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In: Proceedings of the ACL Student Research Workshop, ACL 2005, pp. 43–48 (2005)

34. Rosenthal, S., Farra, N., Nakov, P.: SemEval-2017 task 4: sentiment analysis in twitter. In: Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval 2017 (2017)
35. Salas-Zárate, M.d.P., et al.: Sentiment analysis on tweets about diabetes: an aspect-level approach. Comput. Math. Methods Med. **2017**, 1–9 (2017)
36. Severyn, A., Moschitti, A.: Twitter sentiment analysis with deep convolutional neural networks. In: Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2015, pp. 959–962 (2015)
37. Sokolova, M., Lapalme, G.: A systematic analysis of performance measures for classification tasks. Inf. Process. Manag. **45**(4), 427–437 (2009)
38. Spencer, J., Uchyigit, G.: Sentimentor: sentiment analysis on twitter data. In: The 1st International Workshop on Sentiment Discovery from Affective Data, pp. 56–66 (2012)
39. Stojanovski, D., Strezoski, G., Madjarov, G., Dimitrovski, I.: Finki at SemEval-2016 task 4: deep learning architecture for twitter sentiment analysis. In: SemEval@NAACL-HLT (2016)
40. Taboada, M., Brooke, J., Tofiloski, M., Voll, K., Stede, M.: Lexicon-based methods for sentiment analysis. Comput. Linguist. **37**(2), 267–307 (2011)
41. Turney, P.D.: Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, ACL 2002, pp. 417–424 (2002)
42. Wang, X., Zhang, H., Yuan, S., Wang, J., Zhou, Y.: Sentiment processing of social media information from both wireless and wired network. EURASIP J. Wirel. Commun. Netw. **2016**(1), 146 (2016)

Improvement of Reverse Dictionary by Tuning Word Vectors and Category Inference

Yuya Morinaga(✉) and Kazunori Yamaguchi(✉)

The University of Tokyo, 3-8-1 Komaba, Meguro-ku, Tokyo 153-8902, Japan
{morinaga,yamaguch}@graco.c.u-tokyo.ac.jp

Abstract. A reverse dictionary is a system that returns words based on user descriptions or definitions. OneLook Reverse Dictionary is a commercial reverse dictionary system constructed from existing dictionaries. Hill (2016) reported another reverse dictionary system was constructed from public dictionaries using word embeddings and that its performance was comparable to that of OneLook Reverse Dictionary at the time of the comparison. In this paper we report that, by selecting word vectors suitable for a reverse dictionary and combining Convolutional Neural Network text classification, we improved the reverse dictionary described by Hill. It is very significant that our model can automatically construct a reverse dictionary system from publicly available resources such that it obtains similar scores to those obtained with OneLook Reverse Dictionary in accuracy@100/1000. We also show that our model can be used as a filter to the OneLook Reverse Dictionary to improve its performance.

Keywords: Reverse dictionary · Concept search · Word embedding
Recurrent neural network · Convolutional neural network
Text classification · WordNet

1 Introduction

A dictionary maps a word to its definition while a *reverse dictionary* maps a description to the word specified by the description. Reverse dictionary applications include the tip-of-the-tongue problem [3] and the cross word problem [1].

One of the difficulties in developing a reverse dictionary is that we can not exhaustively enumerate descriptions for a word. For example, the definition of the word 'brother' in WordNet [4] is 'a male with the same parents as someone else', but a reverse dictionary should be able to map the description of 'son of my parents' to 'brother' also. To achieve this, a reverse dictionary should have some mechanism to calculate the similarity between unseen inputs and candidate words.

Several academic studies have proposed reverse dictionary models. Most of previous academic researches on English reverse dictionaries, such as [5,6], use hand-engineered features of sentences. On the other hand, [1] proposed reverse

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 533–545, 2018.
https://doi.org/10.1007/978-3-319-99972-2_44

dictionary model based on Neural Language Model [7], which uses word embeddings as features. Reverse dictionary models of [1,6] are publicly available.

The Node-Graph Architecture proposed in [6] is a reverse dictionary that searches words in a graph where nodes are words and edges are the relations whose source nodes (words) appear in the descriptions of the target nodes (words). In this graph, the length of a path between nodes represents the distance between the nodes and the frequency of words represents the importance of the words. Outputs are the words that are closest to the important input words. This reverse dictionary was claimed to outperform OneLook Reverse Dictionary[1] when the vocabulary is limited to 3,000 words. However, the performance for a larger vocabulary is not available.

The Reverse Dictionary using Word Embeddings [1] (RDWE for short) can handle a larger vocabulary. This reverse dictionary converts the word vectors of word2vec for an input description into a vector by a linear transformation or a Recurrent Neural Network. This reverse dictionary was claimed to have the performance similar to that of OneLook Reverse Dictionary at the moment of [1] in an evaluation using the descriptions provided by users.

In this paper we improved the performance of the reverse dictionary through the use of two techniques. First, we generated word vectors more suited to the use of a reverse dictionary by using the dictionaries with non-ASCII characters removed. Second, following in the Watson [8] which is the Question Answering system employs category inference of answer, we employed a category information to eliminate similar but irrelevant vectors. We found that words belonging to different categories may have similar word vectors. This causes the reverse dictionary to provide incorrect words in irrelevant categories. This problem can be alleviated by filtering out words in the irrelevant categories.

Experiment results we obtained showed that the present OneLook Reverse Dictionary performs better than the reverse dictionary reported in [1] and outperformed RDWE including this work in some metrics. However, RDWE results are quite different from those returned by OneLook Reverse Dictionary, although the rate at which the target word is included in the top 100–1000 words is similar in both dictionaries. Accordingly, we were able to use RDWE as an output filter of OneLook Reverse Dictionary and in so doing we confirmed that search performance was improved.

The contributions of this paper are as follows:

- Improved RDWE accuracy by employing better word vectors
- Improved RDWE accuracy by employing category inference
- Improvement over a commercial reverse dictionary by employing RDWE as a filter.

The rest of this paper is organized as follows. The models we used are explained briefly in Sect. 2. Improvement and analysis on word vectors are explained in Sect. 3. We describe our proposed model in Sect. 4. Section 5 shows the experimental results we obtained. Section 6 concludes the paper with a summary of key points.

[1] http://onelook.com/reverse-dictionary.shtml.

2 Preliminary Definitions

In this section, we will briefly explain RDWE in Sect. 2.1 and a text classification model in Sect. 2.2.

We will denote the vector for word w as v_w and a description as $s = (w_1, w_2, ..., w_n)$.

2.1 Distributed Representation of Sentences

The RDWE outputs words sorted by rank determined by the cosine similarity between the vector of an input description and the vectors of the word. RDWE based on Neural Language Model [7].

Word vectors may be learned independently from a corpus or learned simultaneously in adjusting RDWE parameters. [1] reported that there was almost no difference in the results obtained for either case. In our work, we therefore used word vectors learned from a corpus.

word2vec Add (ADD). We call the model to use the sum of the word vectors of words in a description as *word2vec add model* (ADD for short). If word vectors are given, no learning is needed in ADD. In the work we describe in this paper, we used ADD as the baseline. In experiments we performed, we omitted word vectors of stop words in the summation although they were not omitted in [1]. This omission improves the accuracy.

Bag of Words (BOW). The *Bag of Words* distributed representation model (*BOW* for short) outputs a linearly transformed summation of the vectors of input words as follows. Here, D is the dimension of v, W is a matrix of $D \times D$, and A_j is a D dimensional vector.

$$A_1 = W v_{w_1},$$
$$A_{i(>1)} = A_{i-1} + W v_{w_i} (= W \sum_{j=1}^{i} v_{w_j}).$$

Because W is a linear transformation, A_n is equal to the summation of vectors of the words in a description multiplied by W. The order of words is neglected in this model. We learn W by the stochastic gradient method minimizing the cost between the vector thus calculated and the correct word vector available in the training data. As a cost function, the cosine distance and the rank loss function are used. The rank loss function is determined as follows.

$$\max(0,\ 0.1 - \cos(v_s, v_t) + \cos(v_s, v_{\mathrm{rand}})),$$

where s is a description, t is the correct word, v_s is the vector for s, v_{rand} is a randomly selected vector not identical to v_t, and $\cos(a, b)$ is the cosine similarity function.

Recurrent Neural Network (RNN). *Recurrent Neural Network distributed representation model* (*RNN* for short) inputs words in a description one by one to calculate the next vector and outputs the final vector. Here, A_j is the vector after the j-th word in a description is processed.

$$A_1 = \phi(Wv_{w_1} + b),$$
$$A_{i(>1)} = \phi(UA_{i-1} + Wv_{w_i} + b),$$

where W and U are $D \times D$ matrices, A_j is a D-dim vector, and $\tanh(x)$ is used as an activation function $\phi(x)$.

A_n is the final distributed representation representing the description $(w_1, w_2, ..., w_n)$. In this model, the order of words has an effect on the final output. W, U, and b are learned by the learning method and the cost function for RNN. LSTM (Long Short Term Memory) [9] is employed for long backpropagation.

2.2 Text Classification Model

Reverse dictionary can be regarded as a variety of Question Answering (QA) system, therefore QA system's technique can be applied to. IBM's Watson [8] is a well-known QA system that won against two of *Jeopardy*'s greatest champions in 2011. It classifies question, and judges appropriate answer type [10]. Watson's architecture is proprietary knowledge and specialized in *Jeopardy*, but still we can take similar approach by building another category inference system.

In the following, we will explain a text classification model we employed for our category inference system.

Convolutional Neural Network (CNN). *Convolutional Neural Network text classification model* (*CNN* for short) [2] is used to infer a category to which a word with the description belongs.

In CNN, C is the characteristic vector and p_i is the probability with which s belongs to category i for a given input s, $v_{w_{j:j+h-1}}$ is an $h \times D$ matrix of the concatenation of w_j to w_{j+h-1} of D-dim vectors, and W_i is a $h \times D$ matrix.

$$c_{i,j} = \phi(W_i v_{w_{j:j+h-1}} + b_i),$$
$$c_i = \max_j(c_{i.j}),$$
$$C = (c_1, c_2, ..., c_k),$$
$$p_i = \frac{\exp(c_i)}{\sum_m \exp(c_m)}.$$

This transforms a word sequence into features and the features c_j are used to yield the probability of the i-th category by softmax.

We adjusted the parameters by using the stochastic gradient on the cost determined by the inferred probability and that in the training data. In order to avoid the overfitting, the dropout (masking W randomly when learning) was employed.

3 Word Vectors

3.1 Improving Word Vectors

The performance of RDWE depends on the word vectors it uses. The purpose of this section is to get word vectors suited to RDWE. We measured the suitability of word vectors by using ADD, i.e., the score obtained with the ADD reverse dictionary using the word vectors.

Note that we eliminated the stop words from the search space and overlooked stop words in the summation process. The target words were limited to those in the WordNet vocabulary.

There are two approaches to improve the word vectors. The first is to modify the corpus to use in the word2vec learning such as the tokenization of frequent phrases [11]. The second is to modify the model of word2vec such as the Multi-Sense Skip-gram model [12]. Because the second approach has potential problems with increased learning time and search space, we used the first approach.

Table 1. ADD scores. HIL uses word vectors in the Hill's distribution. AOC uses word vectors freshly learned from AOC.

Word vector	accuracy@1/10/100	Median
HIL	**0.02**/0.156/0.39	213
AOC	0.018/**0.192**/**0.454**	**143**

The results are shown in Table 1. The scores were calculated for the labeled data by users included in the Hill's data. We used the 23 GB text data collected from the Internet (including Wikipedia)[2] as the corpus.

HIL is the Hill's corpus including sentences with non-ASCII characters. The sentences are written in languages other than English and so they constitute noise for learning English word vectors. In total, 16 GB were left when we removed sentences which include non-ASCII characters from the corpus. We call the resulting corpus as *ASCII only corpus* (*AOC* for short).

Word vectors were learned from these corpora and their performances were measured by ADD. The test was conducted on the 500 samples obtained from WordNet. We used the accuracy@1/10/100, which is the rate at which the target word is in the top 1/10/100 word(s) and the median of the target word rank according to the rank ADD assigned for the word vectors.

The purpose of this learning of word vectors is to obtain word vectors suited for the reverse dictionary used for users' descriptions. Therefore, the selection of word vectors has to be done using users' descriptions. However because the number of available users' descriptions was small (200), all of them were used for the final evaluation and the selection of the word vectors was done by using

[2] Collected in the first part of https://github.com/svn2github/word2vec/blob/master/demo-train-big-model-v1.sh.

dictionary data. Each sample consists of a description and the word described by the description. We removed the part enclosed by parentheses because the enclosed part often includes the defined word. The results are shown in Table 1. The p-values of accuracy@1/10/100 and median were respectively 0.617, 0.067, 0.017, and 0.076. This means that our newly learned word vectors using AOC are statistically significantly better at accuracy@100.

3.2 Characteristics of Word Vectors

word2vec learns vectors of words from their context words. As a result, words with similar context have similar vectors in terms of the cosine similarity. However, they do not necessarily belong to the same 'category.' For example, 'bed' and 'asleep', 'cat' and 'meow', or 'mud' and 'muddy' have the similar context resulting in the similar word vectors. This is not desirable for word vectors in reverse dictionary applications because when the input sentence is something like 'an animal that...' it is obvious that the category of the target word is an animal (noun), but the search result may include 'meow' (verb).

In the work we report in this paper, we employed the lexname in WordNet as a category. A lexname is one of 45 kinds of tags determined for each synset such as noun.animal, verb.emotion, and so on. Note that one word may belongs to multiple synsets and, as a result, have multiple lexnames.

Figure 1 shows the distribution of word vectors visualized in 2-dimensional MDS using the cosine distance between word vectors. From this figure, we see that the word vectors of these two categories overlap considerably. Roughly speaking, one third of the target words do not have any common categories with more than half of the nearest 100 words of the words. As this shows, a considerable number of words close to the calculated word vector fall in wrong categories.

4 Proposed Model

The distributed representation of the target word is calculated from the distributed representation of the words in an input sentence as in [1]. As shown in Sect. 3.2, words similar to the distributed representation may belong to wrong categories. By using the category of the target word, we can eliminate the obviously irrelevant words from the ranked result and improve the accuracy. Even though the category of the target word is not given, we can infer its category. In our work, we employed the CNN [2] for this.

The CNN result is the distribution on categories. If we use a small number of categories, the chance that the correct category is included is small. Once it is included the rank of the target word in the ranked result is high, but otherwise, we will miss the target word in the ranked list. Therefore, there is a trade-off between the precision and recall. We propose to control the trade-off by the estimated probability of selected categories. Let us assume that the probabilities of the target word t_q of an input sentence q belongs to the categories $c_1, c_2, ..., c_n$

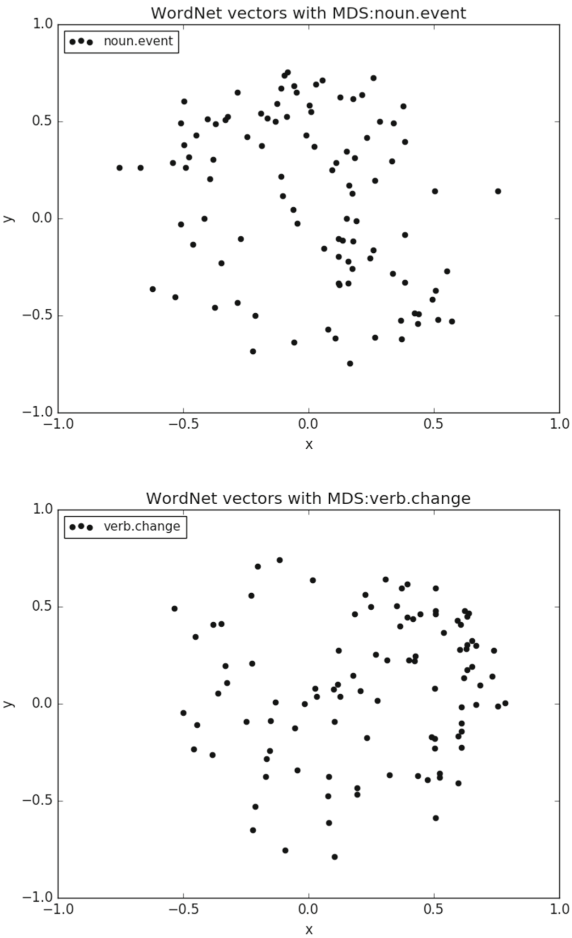


Fig. 1. 2-dimensional MDS of word vectors: 100 sample words each in noun.event (above) and verb.change (below).

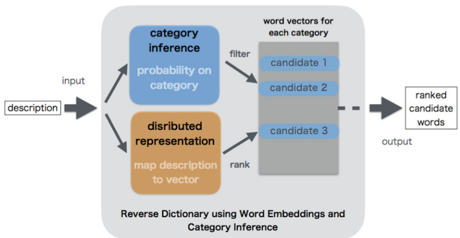


Fig. 2. Reverse dictionary using word embeddings and category inference (RDWECI) (our proposal)

are $p_{c_1} \geq p_{c_2} \geq ... \geq p_{c_n}$[3]. Let us also assume that the probabilities belonging to categories are independent. Then, the probability that t_q does not belong to any of the top m categories is $\prod_{i=1}^{m}(1-p_{c_i})$. By employing m_0 categories where $m_0 = \underset{m}{\arg\min} \prod_{i=1}^{m}(1-p_{c_i}) < k$, we can keep this *failure rate* as low as k. Finally, we rank only words which are in the m_0 categories.

Our algorithm is summarized below.

Input: input sentence q, failure-rate k.
1. Calculate the distributed representation of v_q of q.
2. Calculate the probabilities $p_{c_1^q} \geq p_{c_2^q} \geq ... \geq p_{c_n^q}$ on categories with which the target word of q belong to the categories.
3. Determine the top m_0 categories $c_1^q, ..., c_{m_0}^q$ for k.
4. Rank words belonging to the categories $c_1^q, ..., c_{m_0}^q$ according to the cosine similarity with the word vector of the words and v_q.
Output: Top 1/10/100 word(s) in the ranked words.

The whole architecture is shown in Fig. 2. We call this model *RDWECI*.

5 Experiment

5.1 Data

Training Data. For training Hill's reverse dictionary model, we used the word and description pairs obtained from Wordnik API[4]. We accessed the word and description pairs from Wordnik API according to a word list distributed by Hill, removed remaining HTML tags such as "`<strong>`" from the obtained data. The results were about 540,000 word and description pairs and used in Sect. 5.2.

For training the text classification model, we used the lexname and definition pairs for each synset of WordNet. The results were about 120,000 category and definition pairs and is used in Sect. 5.3.

As to word vectors, we used HIL and AOC in Sect. 3.

Test Data and Evaluation Metrics. For the evaluation, we used the 200 word and description pairs in the data distributed by Hill.

For evaluation metrics, we used accuracy@1/10/100 and median.

Because we could search only a word whose category is defined in WordNet, our search space included the 90,000 words of the WordNet whose word vectors were available. This is broader than the 66,000 words space used by Hill.

Because about 80 words in the Hill's corpus are not included in AOC, the search space was slightly different between HIL and AOC.

For evaluating the text classification model, we used the above data and WordNet's seen and unseen data. The WordNet's data consists of the lexname

[3] In our experiment, $n = 45$.
[4] http://developer.wordnik.com.

and definition pairs for each synset of WordNet in Sect. 5.1. In calculating the accuracy, we considered the result was correct when the top category inferred was included in the categories to which one of the synsets of the word is assigned in WordNet. If the WordNet data is used, because the description defines a synset, we used the category of synset as the correct one. One definition in the WordNet corresponds to one synset having one category (lexname). Each of Hill's data is a pair of a word and its description, and the word may belong to multiple synsets and categories. Thus, for the evaluation using the Hill's data, we considered the result was correct if one of the categories to which the target word belongs is named.

5.2 Reverse Dictionary Model

Training. The reverse dictionary model is trained as Sect. 2.1. As to word vectors, we used those in HIL and AOC as in Sect. 3.1. We tried two types of networks: BOW and RNN, and two types of cost functions: cosine and rankloss. We used the implementation by Hill[5].

Evaluation. In this section, we conducted the evaluation of the trained reverse dictionary model by the data in Sect. 5.1. The results are shown in Figs. 3 and 4 in Sect. 5.4.

In general, the scores for AOC were better than those for HIL. As to networks and cost functions, BOW rankloss was the best. RNN was better in accuracy@1, but not so for other metrics. This may be the result of the overfitting by RNN which is more complex and flexible than BOW.

5.3 Text Classification Model

Training. As to CNN in Sect. 2.1 we used the implementation in the Harvard NLP[6]. As to word vectors, we used HIL and AOC as in Sect. 3.1.

Evaluation. The accuracy of the trained model was 93% (HIL)/94% (AOC) for the trained WordNet definitions, 83% (HIL)/84% (AOC) for the WordNet definitions not included in the training data, and 70% (HIL)/70% (AOC) for the Hill's evaluation data.

5.4 Evaluation of RDWECI

We evaluated RDWECI proposed in Sect. 4 by combining the models in Sects. 5.2 and 5.3. The results are shown in Figs. 3 and 4.

[5] https://github.com/fh295/DefGen2.
[6] https://github.com/harvardnlp/sent-conv-torch.

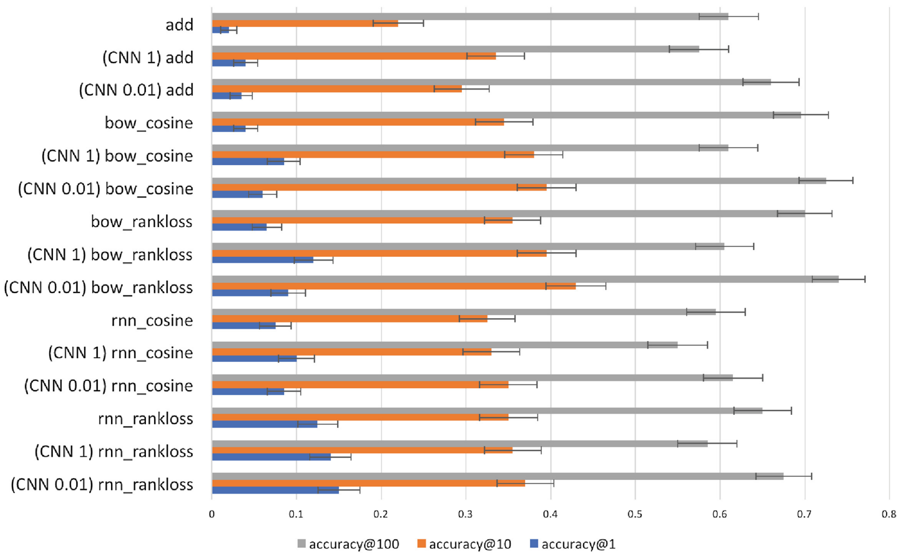


Fig. 3. Accuracy for various models and parameters for AOC. CNN is our model and others are Hill's model. Error bars show the standard deviations.

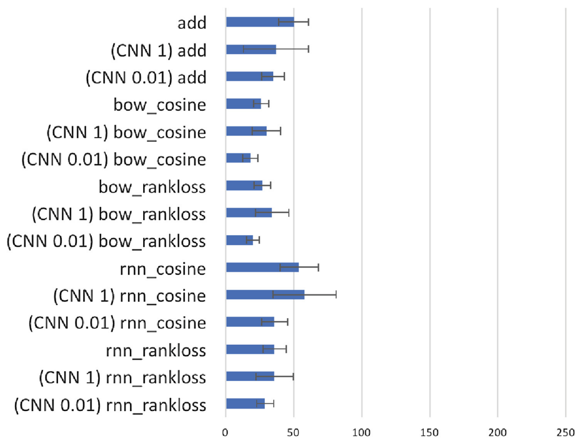


Fig. 4. Median for various combinations and parameters for AOC. CNN is our model and others are Hill's model. Error bars show the standard deviations.

For the failure-rate in Sect. 4, we tried 1 and 0.01. This failure-rate was chosen as follows. We tested the model with failure-rate = 1/0.5/0.4/0.3/0.2/0.1/0.05/0.01/0.001 using the dictionary data and the best score is obtained at failure-rate = 0.01. Therefore, in the following, we will use 0.01 as the failure-rate. For comparison, we use 1 as the failure-rate, because with failure-rate = 1 only the top category is used and this is the other extreme. In order to avoid overfitting, we didn't use the test data for this tuning.

In the realization of the Hill's experiment in [1], BOW rankloss using Hill's distributed data performed best resulting in accuracy@1/10/100 = 0.06/0.325/0.61 and median = 50. According to RDWECI, CNN 0.01 BOW rankloss AOC performed best marking accuracy@1/10/100 = 0.09/0.43/0.74 and median = 20. The difference between Hill's model and ours best are statistically significant in accuracy@10 ($p = 1.59 \times 10^{-2}$), accuracy@100 ($p = 2.20 \times 10^{-3}$), and median ($p = 4.40 \times 10^{-3}$).

5.5 OneLook Reverse Dictionary

Evaluation of OneLook Reverse Dictionary. OneLook Reverse Dictionary is the commercial reverse dictionary system. One can search words from their description from 1,061 dictionaries.

The score of the OneLook Reverse Dictionary was reported as accuracy@10/100 = 0.38/0.58[7] and median = 18.5, similar to the performance of Hill's BOW/RNN models.

We used data included in Hill's distribution in Sect. 5.1 for this and the next evaluation.

The scores of our RDWECI were higher than those of Hill's BOW/RNN model, and the OneLook Reverse Dictionary at the moment of [1].

We evaluated the score of the OneLook Reverse Dictionary on Oct.16, 2017. The results were accuracy@1/10/100 = 0.34/0.55/0.76 and median = 5 and accuracy@1/10/100 = 0.33/0.54/0.76 and median = 6.

OneLook Reverse Dictionary Enhanced by the Distributed Representation and Category Inference. OneLook Reverse Dictionary and RDWECI have different architectures and only 10–15% words are common in the top 1000 words lists of them. OneLook Reverse Dictionary performs well for accuracy@1/10 but RDWECI performs relatively well for accuracy@100/1000; the accuracy@100/1000 of OneLook Reverse Dictionary is 0.76/0.87 and that of RDWECI is 0.74/0.89.

From this observation, we constructed the following algorithm to combine the bests.

Input: Input sentence q.
1. Calculate the top 1,000 words m by RDWECI for q.
2. Search the top 1,000 words r by OneLook Reverse Dictionary for q.
3. Remove words from r which are not in m.
Output: r

As shown in Table 2, the combination improved the score.

In this combination, the BOW and RNN did not perform any better than ADD. This may be because ADD is not so flexible in providing the target word at a higher rank, but it is robust enough to keep it in the top words list.

[7] accuracy@1 was not reported in [1].

Table 2. OneLook Reverse Dictionary Enhanced by the Distributed Representation and Category Inference (1) ADD, (2) CNN 0.01 add, (3) BOW rankloss, (4) CNN 0.01 BOW rankloss. All word vectors are from AOC.

Model	accuracy@1/10/100	Median
OneLook	0.34/0.55/0.76	5
(1) + OneLook	0.36/0.62/0.81	4
(2) + OneLook	**0.38/0.63/0.82**	**3**
(3) + OneLook	0.35/0.60/0.80	4
(4) + OneLook	0.36/0.61/**0.82**	**3**

6 Conclusion

In the work we described in this paper, we improved the accuracy of the reverse dictionary of [1] by selecting better word vectors and introducing category inference. Our best model performs similarly to OneLook Reverse Dictionary according to accuracy@1000. This is very significant because OneLook Reverse Dictionary is a commercial and dedicated system while our system is built using only category information as natural language knowledge and public dictionaries. We showed that we can also use RDWECI as a filter to improve the OneLook Reverse Dictionary. Various subjects for future work remain. First, we would like to use multi-sense word vectors in the reverse dictionary. By capturing more precise meanings of words by using such vectors in an input sentence, we may be able to improve reverse dictionary performance. Second, we would like to build a reverse dictionary model more suitable to be used as filter of OneLook Reverse Dictionary. As a filter to the OneLook Reverse Dictionary, our model with category inference shows only a marginal improvement to ADD. By employing a more suitable function to improve accuracy@1000, we may find a filter more suited to OneLook Reverse Dictionary.

References

1. Hill, F., Cho, K.H., Korhonen, A., Bengio, Y.: Learning to understand phrases by embedding the dictionary. Trans. Assoc. Comput. Linguist. **4**, 17–30 (2016)
2. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1746–1751 (2014)
3. Schwartz, B.L., Metcalfe, J.: Tip-of-the-tongue (TOT) states: retrieval, behavior, and experience. Mem. Cogn. **39**(5), 737–749 (2011)
4. Miller, G.A.: Wordnet: a lexical database for english. Commun. ACM **38**(11), 39–41 (1995)
5. Shaw, R., Datta, A., VanderMeer, D., Dutta, K.: Building a scalable database-driven reverse dictionary. IEEE Trans. Knowl. Data Eng. **25**(3), 528–540 (2013)
6. Thorat, S., Choudhari, V.: Implementing a reverse dictionary, based on word definitions, using a node-graph architecture. In: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics, pp. 2797–2806 (2016)

7. Bengio, Y., Ducharme, R., Vincent, P., Jauvin, C.: A neural probabilistic language model. J. Mach. Learn. Res. **3**(Feb), 1137–1155 (2003)
8. Ferrucci, D., et al.: Building Watson: an overview of the DeepQA project. AI Mag. **31**(3), 59–79 (2010)
9. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
10. Lally, A.: Question analysis: how Watson reads a clue. IBM J. Res. Dev. **56**(3.4), 2:1–2:14 (2012)
11. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
12. Neelakantan, A., Shankar, J., Passos, R., Mccallum, A.: Efficient nonparametric estimation of multiple embeddings per word in vector space. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1059–1069 (2014)

Determining Quality of Articles in Polish Wikipedia Based on Linguistic Features

Włodzimierz Lewoniewski(✉), Krzysztof Węcel, and Witold Abramowicz

Department of Information Systems, Poznań University of Economics and Business, al. Niepodległości 10, 61-875 Poznań, Poland
{wlodzimierz.lewoniewski,krzysztof.wecel,witold.abramowicz}@ue.poznan.pl
http://kie.ue.poznan.pl

Abstract. Wikipedia is the most popular and the largest user-generated source of knowledge on the Web. Quality of the information in this encyclopedia is often questioned. Therefore, Wikipedians have developed an award system for high quality articles, which follows the specific style guidelines. Nevertheless, more than 1.2 million articles in Polish Wikipedia are unassessed. This paper considers over 100 linguistic features to determine the quality of Wikipedia articles in Polish language. We evaluate our models on 500 000 articles of Polish Wikipedia. Additionally, we discuss the importance of linguistic features for quality prediction.

Keywords: Wikipedia · NLP · Data Quality · Quality Assessment Random Forest · Polish · Linguistic Features · Linguistics Data Mining

1 Introduction

Information has become a commodity. The quantity and quality of information largely determine the quality of decisions in various business branches. On the one hand, managers care about having access to as wide range of information as possible. On the other hand, the quality of information determined by different features (such as relevance, accuracy, unambiguity and others) is also important.

Nowadays everyone can contribute to common human knowledge on the Web. One of the best examples of such online repositories is Wikipedia, which have more than 46 million articles in 288 active language editions [21]. Polish version of this encyclopedia is one of the biggest language editions. As of March 2017, the Polish-language edition of Wikipedia had more than 1.2 million articles and more than 250 million page views per month.

In Wikipedia there are systems of grades for article quality and particular language version can use own assessment standard [14,20]. Each language version have special awards for articles with the best quality. In English version such articles are called "Featured articles" (FA). In Polish Wikipedia articles with the highest quality have name"Artykuły na medal", what is essentially equivalent

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 546–558, 2018.
https://doi.org/10.1007/978-3-319-99972-2_45

to FA grade in English. Such articles should be well written, in particular fulfill certain criteria: encyclopedic style, uniformity of tense, neutrality and other [23]. The language of encyclopedia article must be very precise. It is necessary to avoid clutter of thoughts or excessive coloration. Separate rules draw attention to lead section of article: it should be clear, not very long or detailed, written in accessible language, allowing the reader to learn the content without reading the whole article [22]. Articles that meet a core set of editorial standards but are not featured articles, qualify as "Good articles" (GA); in Polish language – "Dobre artykuły".

Usually in each language version of Wikipedia there are only about 0,4–0,6% of high-quality articles (marked as FA or GA). Other articles can get lower quality grades. However, in some developed language versions of Wikipedia still most of the articles are unevaluated. For example, in Polish Wikipedia the share of articles without quality grade is about 99%. This number could be lowered by involving more experienced users and experts from different disciplines. Unfortunately, such experts are not always available.

Writing style of articles depends on the language characteristics. FA and GA articles cover more concepts, objects and facts than other ones [15]. So we can expect, that these articles used more nouns and verbs and less adjectives [31]. There are different studies that estimate the quality of articles in English Wikipedia based on various lexical features. However, there are no such studies related to Polish Wikipedia. Even more, there are no studies examining the use of specific Polish linguistic characteristics to evaluate the quality of texts in general.

In this paper we use over 100 linguistic features to determine quality of articles in Polish Wikipedia. Apart from the entire text of each of the article, we also analyzed features of a special part – the leading section. Our model achieved high precision and we test our model on selected unevaluated articles to predict the quality. This work is the first research, which used Polish linguistic features to assess quality of Wikipedia articles.

2 Related Work

Existing studies describes various ways to predict the quality of the Wikipedia articles. Some of them determine the quality based on article's content, another uses the edit history, the article's talk page and other sources. In general, we can divide related studies into the two groups: content-based and user-based approaches. Existing research works proposed different feature sets for measuring quality of Wikipedia articles.

Works related to user's (editor's) behavior, explore how the users experience and coordinate their activities in relation to article quality. These approaches use various characteristics related to a user reputation and changes that they made [17,30]. Usually high quality articles have a large number of editors and edits [27]. Interaction among editors and articles can be visualized as a network, and using graph theory structural features associated to articles quality can be determined [11]. There is also artificial intelligence service involved to discover damaging edits, which can be used to immediately score the quality [10]. However, such user-based methods often require complex calculations and they do not analyze article itself, which would indicate what needs to be changed to improve its quality.

Another group of the scientific works analyzed the article content. One of the first studies showed that longer articles in Wikipedia often had higher quality grades [4]. Later works identified other features related to various constituents of the article: the best articles have more images, sections, use bigger number of references than articles with lower quality [13,19,20]. Online service WikiRank [25] use some of the content quantitative features to assess relative quality of Wikipedia articles in different languages [14]. Special quality flaw templates can also help in articles assessment in Wikipedia [3]. A few works try to combine features from edition history and articles content [6,7]. Other study used Deep Learning approaches to tackle the quality classification problem and compared results with recent developments [8].

Recently, in scientific works more and more attention is paid to analyzing not only the quantitative features of the text of articles but also qualitative. One of the studies used character trigram feature to analyze article writing style, which can be a predictor for its quality [16]. Another study used 8 basic lexical metrics derived from the statistic on word usages in Wikipedia articles as the factors that can reflect its quality [31]. Linguistic features can also be used to examine how density of factual information impacts articles quality: bigger relative number of facts in a document indicates its more informative [15].

Concluding, existing studies propose different feature sets for assessing quality of articles in Wikipedia. However, there is no single universal feature set for doing it [7], especially if we consider different language versions [13,20]. Nowadays, Wikipedia contains articles in over 290 languages [21]. Most of the research related to automatic quality assessment based on linguistic parameters is associated with the biggest language version of Wikipedia - English [15,16,31], which has more than 5.5 million articles. The most similar work for this study was done recently for articles in Russian Wikipedia [12]. However, we did not find such studies related to Polish language.

Our work is related to content-based approach and using more than 100 linguistic features to predict the quality of Polish articles in Wikipedia. In addition to the entire articles text, we decided to separately explore the linguistic features of one of its most important parts - leading section.

3 Preparing the Dataset

In Polish Wikipedia there are over 2300 GA and over 700 FA articles, which cover topics such as: biographies, humanities, arts, social sciences, earth sciences, mathematics, physical sciences, chemical sciences, biological sciences, medical sciences, technical sciences, military, and others. Depending on the language version, Wikipedia articles can have other (lower) quality grades [13,14,20].

Table 1. Articles count with quality grade in Polish Wikipedia. Source: own calculations in May 2017.

Quality grade	Articles count
Featured article	723
Good article	2 303
Correct	785
Start	1 246
Stub	1 635
Without grade	1 212 559

In the case of the Polish Wikipedia there are additional 5 quality grades, which show the degree of development of the article: four, correct, sufficient, start, stub. In Polish Wikipedia particular projects can use different names of quality grades. However, we can find similarities between some grades when comparing their criteria. As a result, we group them to 3 lower quality grades: correct (four), start (sufficient), stub. The number of articles with certain quality grade can be seen in Table 1.

According to previous works [7,12,13,15,19,20], we divide all grades into two classes:

- Complete articles (with FA and GA).
- To-improve (with Correct, Start, Stub).

To train our model we also decided to use two sampling methods:

- Imbalanced - with all assessed articles in each class.
- Balanced - with equal number of randomly extracted articles in each class (stratified sampling).

In case of Balanced dataset, the number of articles of each grade was based on maximum articles count in the smallest class (Complete), and taking into account the equal representation of each grade within each class. Imbalanced dataset used all the available evaluated articles. Details are presented in Table 2.

Table 2. Articles count in each class in two data sets

Quality grade	Imbalanced	Balanced	Class
Featured article	723	720	Complete
Good article	2 303	720	Complete
Correct	785	480	Uncomplete
Start	1 246	480	Uncomplete
Stub	1 635	480	Uncomplete
Total	**6 692**	**2 880**	

4 Features Extraction

Wikipedia have special web service that provides convenient access to wiki features, data, and meta-data over HTTP. We use it to extract the texts of the articles (with leading section). We implemented own application to extract linguistic features based on morphological vocabulary - PoliMorf [28]. Most of the parameters have the notation according to morphosyntactic marker system used in the corpus of Institute of Computer Science, Polish Academy of Sciences [29]. Moreover, Polimorf includes inflections as well as word formations, therefore it was also used for stemming.

Aside from the standard markup for the Polish language, we included additional features such as: number of unique words[1] (and separately verbs, nouns, adjectives), noun to verb ratio, long words (with over 3 syllables) and others. Some of these additional features are also used in assessing the readability of Polish texts [9]. List of analyzed features can be found in Table 3.

We also used frequency lists extracted from large texts corpora, including National Corpus of Polish Language, Rzeczpospolita (newspaper), Polish Wikipedia and others. These corpora have about 1.8 billion tokens in common and SuperMatrix utilities were used to generate the frequency list [5]. We created features that count separately words from top 50, 100, 200, 300, 400, 500 and 1000 words of frequency list of base forms used in each article (*f50, f100, f200, f300, f400, f500, f1000*). We did similarly for popular words taken from frequency list of Polish Wiktionary [26] (*w50, w100, w200, w300, w400, w500, w1000*). Before the feature calculation, we converted each word in articles into the base form using the PoliMorf vocabulary mentioned before.

Previous works showed that articles with high quality usually have more text (length, number of words) [4,14,16]. Therefore we can expect that the value of a majority of analyzed linguistic features will correlated with the quality of the article - the larger the article the bigger will be the number of nouns, verbs and other parts of speech. Therefore, we decided to normalize all features by words count. In this case, the density of these parts of speech will play a greater role. We expect predicting quality with such normalized features be a challenging task.

We have analysed the values distribution of various features in relation to quality classes. It is basically not possible to unambiguously assign article to the class, based only on one feature. Distribution of some features is presented in Fig. 1.

In addition, we took into account features from only leading sections of a Wikipedia articles. This sections placed before the table of contents and the first heading, serves as an introduction to the article and a summary of its most important contents. Wikipedia community have separate rules for writing the leading section [24].

[1] Unique words was counted on base forms each of each in the texts.

Table 3. List of analyzed linguistic features. Source: own work.

Name	Description	Name	Description
acc	accusative case	*nie*	words with "nie" (negation)
add	additional term	*nom*	nominative case
adj	adjective	*noun*	subst + depr
adja	adjectival adjective	*noun/verb*	noun to verb ration
adjc	predicative adjective	*npraep*	non-post-prepositional
adjp	post-prepositional adjective	*num*	numeral
adv	adverb	*nwok*	non-vocalic
aff	affirmative	*org*	organization
agl	agglutinative	*osoba*	person
aglt	agglutinate form "być	*own*	proper name
akc	accented	*p1*	personal plurale tantum gender
AWLS	avg. words lengths (in syllabs)	*p2*	second plurale tantum gender
bedzie	future form być	*p3*	third plurale tantum gender
burk	bound word	*pact*	active adj. participle
com	comparative degree	*pant*	anterior adv. participle
comp	subordinating conjunction	*pcon*	contemporary adv. participle
congr	agreeing accommodability	*perf*	perfective
conj	coordinating conjunction	*pl*	plural
dat	dative case	*pos*	positive degree
depr	depreciative noun	*posp*	common words
etn	etnonim	*ppas*	passive adj. participle
f	feminine	*ppron12*	non-3rd person pronoun
f100	top 100 of frequency list	*ppron3*	3rd person pronoun
f1000	top 1000 of frequency list	*praep*	post-prepositional
f200	top 200 of frequency list	*praet*	l-participle
f300	top 300 of frequency list	*prd*	product
f400	top 400 of frequency list	*pred*	predicative
f50	top 50 of frequency list	*prep*	preposition
f500	top 500 of frequency list	*pri*	first person
fin	non-past form	*qub*	particle-adverb
gen	genitive case	*rec*	governing accommodability
geo	geographical name	*sec*	second person
ger	gerund	*sg*	singular
imperf	imperfective	*sie*	word "się"
imps	impersonal	*sname*	surname
impt	imperative	*subst*	noun
inf	infinitive	*sup*	superlative degree
inst	instrumental case	*ter*	third person
interj	interjection	*uadj*	unique adjective count

(*continued*)

Table 3. (*continued*)

Name	Description	Name	Description
ladj	long adjectives	*unoun*	unique noun count
lnoun	long nouns	*uverb*	unique verb count
loc	locative case	*verb*	verb
lverb	long verbs	*voc*	vocative case
lword	long words	*w100*	top 100 of wiki-frequency list
m1	personal masculine gender	*w1000*	top 1000 of wiki-frequency list
m2	animate non-personal masculine gender	*w200*	top 200 of wiki-frequency list
m3	inanimate masculine gender	*w300*	top 300 of wiki-frequency list
n1	first neuter gender	*w400*	top 400 of wiki-frequency list
n2	second neuter gender	*w50*	top 50 of wiki-frequency list
nagl	non-agglutinative	*w500*	top 500 of wiki-frequency list
nakc	non-accented	*winien*	word "winien"
name	name	*wok*	vocalic
neg	negative	*wyd*	event

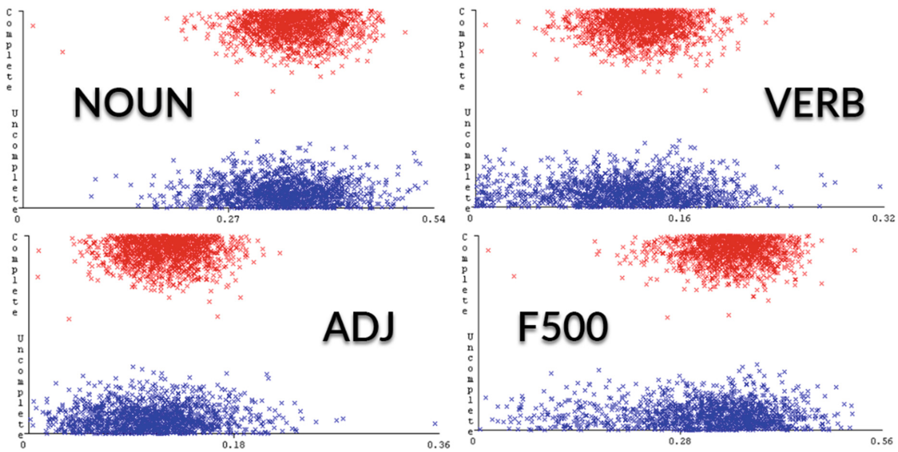


Fig. 1. Feature values (normalized by word count) by quality classes (x-axis shows feature values, y-axis - quality class) in Balanced dataset: nouns (NOUN), verbs (VERB), adjectives (ADJ), words from frequency list F500 - first 500 words. Source: own work using WEKA software.

5 Building the Quality Models

We decide to use Random Forest algorithm to analyze the described linguistic features in order to automatically determine quality classes of Wikipedia articles. This data mining algorithm is robust, i.e. tolerates correlated variables and noise, so it shows the highest precision on similar tasks [12,13,19,20]. We applied its implementation in WEKA software [18] using default settings - 100 trees, cross-validation with 10 folds. In order to build a model, we take into account 106 linguistic features as independent variables and the quality class as the dependent variable. Cross-validation allows to test quality models on independent data sets. Classification accuracy of models with various datasets is presented in Table 4.

Table 4. Classification accuracy (weighted avg.) in percent for Balanced and Imbalanced datasets.

Index	Balanced		Imbalanced	
	Lead. sect.	Text	Lead. sect.	Text
TP rate	81.3	91.8	81.5	93.1
FP rate	18.8	8.2	18.5	6.8
Precision	81.5	92.0	81.6	93.3
Recall	81.3	91.8	81.5	93.1
F-measure	81.2	91.8	81.5	93.1
MCC	62.8	83.8	62.8	86.4
ROC area	89.4	97.1	90.3	97.5
PRC area	89.3	96.9	90.1	97.3

The evaluation shows that higher precision can be achieved when analyzing the whole text compared to the leading section - difference is of about 10 percent. We also observe that one can achieve a slightly larger value of classification accuracy using imbalanced dataset, which is associated with a large number of training samples.

One of the advantages of using the Random Forest is the ability to identify the most significant features for prediction of quality. In Fig. 2 we show importance of all considered features in scale 0–100 (a higher value indicates a higher importance) in different datasets and different analyzed parts (text and leading section).[2] In Table 5 we show the top 10 most significant features for each dataset and analyzed part.

The results showed that there are differences between the quality models based on features taken from the text and from leading section, but are similar if comparing Balanced and Imbalanced dataset. If we consider leading section of articles, the highest importance in prediction quality are relative quantity of common words, locatives, vocatives, third person words, and ordinal. In case of text features, the most important are relative number of unique nouns, unique verbs, 3rd person verbs, common words, and impersonal verbs.

We see, that these results reflects general accepted rules for writing good articles in Wikipedia. Texts in articles with the higher quality usually desccribe subject more detailed and comprehensive. Therefore text of such articles must

[2] More detailed results in tabular form can be found on the page: http://data.lewoniewski.info/icist2018pl/.

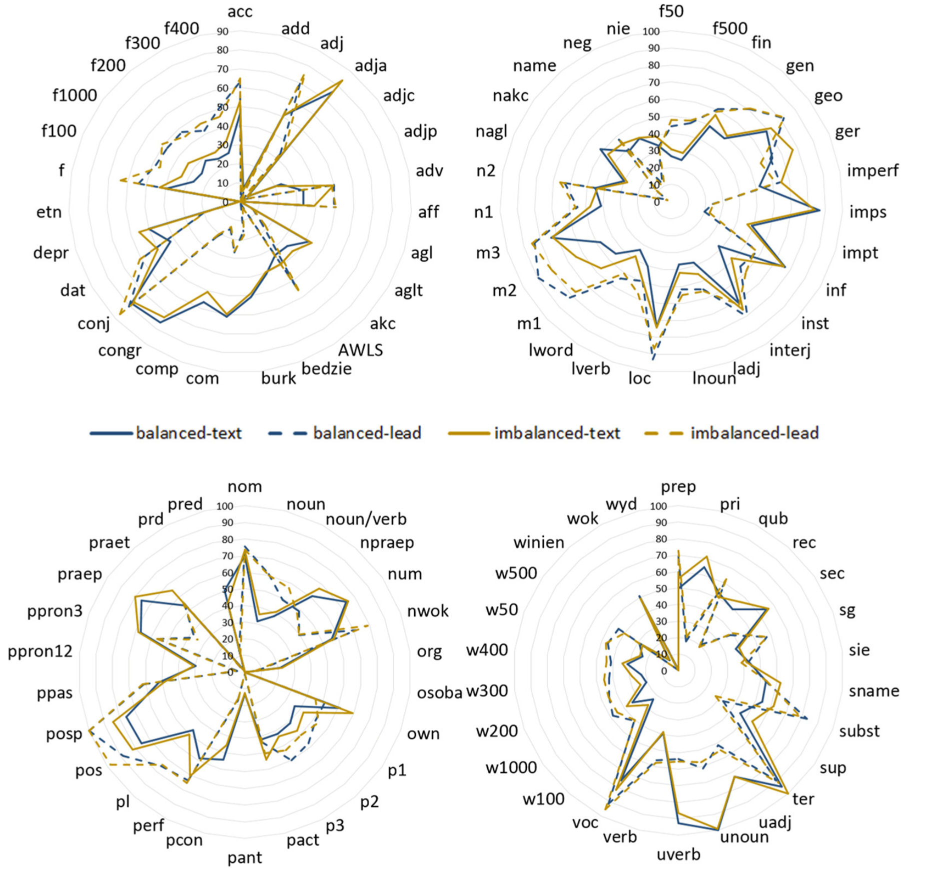


Fig. 2. Feature importance (normalized by number of nodes using that features) in classification model using Random Forest algorithm in different datasets.

consist more unique objects (nouns) and predicates (verbs). Additionaly, articles with higher quality must be written in encoclopedic manner, which obliges to use more words in third person.

On the other hand we have other most important features in leading section, which must present introduction to the article and a summary of its most important contents. That first part of the article must tell the reader the basics and cultivate interest in reading more detailed information (whole article). At the same time, the leading section should be written in a clear and accessible style. Therefore frequency of the common words in such sections can help to discern articles with different quality. Also, in higher quality articles, we can met relatively more grammatical cases which indicates locations and more nouns that indicates persons (vocative).

Table 5. The top 10 most significant features in predicting articles quality in Polish Wikipedia. Source: own calculation using Random Forest algorithm

Balanced		Imbalanced	
Lead. sect.	Text	Lead. sect.	Text
posp	unoun	posp	ter
loc	ter	pos	unoun
voc	uverb	voc	uverb
m2	imps	ter	posp
ter	conj	loc	imps
pos	inf	conj	adja
m3	congr	m3	pos
subst	posp	geo	voc
geo	adja	m2	praep
m1	praep	nwok	conj

6 Articles Evaluation

Quality models based on text features from Balanced and Imbalanced dataset have similar precision. Therefore we used both to assess 500 000 randomly chosen unevaluated articles from Polish Wikipedia. About 4–5% of these articles were determined by both models as Complete. Results are shown in Table 6.

Table 6. Classification results for 500 000 unevaluated articles in Polish Wikipedia using quality models from Balanced and Imbalanced dataset.

Balanced		Imbalanced	
Lead. sect.	Text	Lead. sect.	Text
18 306	481 694	25 209	474 791

Balanced model, which show slightly lower precision than Imbalanced, marks near 3,7% of analyzing articles as Complete. Imbalanced model classified about 5% of these as good quality articles.

7 Conclusions and Future Work

Use of linguistic features is valuable for automatic determination of quality of Wikipedia articles in Polish language. Better results in terms of precision can be achieved when the whole text of article is taken into the account. Then our model shows over 93% classification precision using such features as relative number of unique nouns and verbs (unique, 3rd person, impersonal). However, if we take into account only leading section of an article, relative quantity of common words, locatives, vocatives and third person words are the most significant for determination of quality.

Using the obtained quality models we asses 500 000 randomly chosen unevaluated articles from Polish Wikipedia. According to result, about 4–5% of assessed articles can be considered by Wikipedia community as high quality articles.

We plan to expand the number of considered independent variables in the quality model and complement them with semantic features and readability formulas specific for Polish language. We also plan to increase the number of frequency and topics dictionaries.

Previous work and current study shows that it is possible to build similar models for other languages, so we also plan to compare the quality of information in different language versions of the same article in Wikipedia. However, it requires availability of specialized dictionaries (resources) and language-specific techniques. This is especially challenging task for less developed language editions of Wikipedia such as Belarusian, Ukrainian, Czech and other.

The linguistic features considered in this study can be useful for building more complex quality models of articles in different languages of Wikipedia. This will not only increase the accuracy of determining the quality of articles but also progress from a binary quality classification to a more detailed grading scheme.

In future works we will try to build models for fact extraction from the content of Wikipedia articles in different languages based on linguistic features. It can be useful, for example, to compare the data from Wikipedia infoboxes and statements in raw text of Wikipedia articles. More over, this models can help in future to enrich different semantic databases, such as DBpedia [2], Wikidata [1] and others.

References

1. Dbpedia: Main page. https://wiki.dbpedia.org/
2. Wikidata: Main page. https://www.wikidata.org/wiki/Wikidata:Main_Page
3. Anderka, M.: Analyzing and predicting quality flaws in user-generated content: the case of Wikipedia. Ph.D., Bauhaus-Universitaet Weimar Germany (2013). http://www.uni-weimar.de/medien/webis/publications/papers/anderka_2013.pdf
4. Blumenstock, J.E.: Size matters: word count as a measure of quality on wikipedia. In: WWW, pp. 1095–1096 (2008). https://doi.org/10.1145/1367497.1367673. http://portal.acm.org/citation.cfm?id=1367673
5. Broda, B., Piasecki, M.: Parallel, massive processing in supermatrix: a general tool for distributional semantic analysis of corpora. Int. J. Data Min. Model. Manag. **5**(1), 1–19 (2013)
6. Dalip, D.H., Lima, H., Gonçalves, M.A., Cristo, M., Calado, P.: Quality assessment of collaborative content with minimal information. In: IEEE/ACM Joint Conference on Digital Libraries, pp. 201–210 (2014). https://doi.org/10.1109/JCDL.2014.6970169
7. Dang, Q.V., Ignat, C.L.: Quality assessment of Wikipedia articles without feature engineering. In: 2016 IEEE/ACM Joint Conference on Digital Libraries (JCDL), pp. 27–30, June 2016

8. Dang, Q.V., Ignat, C.L.: An end-to-end learning solution for assessing the quality of Wikipedia articles. In: Proceedings of the 13th International Symposium on Open Collaboration. OpenSym 2017, pp. 4:1–4:10, ACM, New York (2017). https://doi.org/10.1145/3125433.3125448
9. Gruszczyński, W., Broda, B., Charzyńska, E., Dębowski, u., Hadryan, M., Nitoń, B., Ogrodniczuk, M.: Measuring readability of polish texts. In: Proceedings of the 7th Language and Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics, November 2015
10. Halfaker, A., Taraborelli, D.: Artificial intelligence service 'ORES' gives Wikipedians X-Ray specs to see through bad edits (2015). https://blog.wikimedia.org/2015/11/30/artificial-intelligence-x-ray-specs/
11. Ingawale, M., Dutta, A., Roy, R., Seetharaman, P.: Network analysis of user generated content quality in Wikipedia. Online Inf. Rev. **37**(4), 602–619 (2013). https://doi.org/10.1108/OIR-03-2011-0182
12. Lewoniewski, W., Khairova, N., Węcel, K., Stratiienko, N., Abramowicz, W.: Using morphological and semantic features for the quality assessment of Russian Wikipedia. In: Damaševičius, R., Mikašytė, V. (eds.) ICIST 2017. CCIS, vol. 756, pp. 550–560. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67642-5_46
13. Lewoniewski, W., Węcel, K., Abramowicz, W.: Quality and importance of Wikipedia articles in different languages. In: Dregvaite, G., Damasevicius, R. (eds.) ICIST 2016. CCIS, vol. 639, pp. 613–624. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46254-7_50
14. Lewoniewski, W., Węcel, K., Abramowicz, W.: Relative quality and popularity evaluation of multilingual Wikipedia articles. Informatics **4** (2017). https://doi.org/10.3390/informatics4040043, http://www.mdpi.com/2227-9709/4/4/43
15. Lex, E., Voelske, M., Errecalde, M., Ferretti, E., Cagnina, L., Horn, C., Stein, B., Granitzer, M.: Measuring the quality of web content using factual information. In: Proceedings of the 2nd Joint WICOW/AIRWeb Workshop on Web Quality - WebQuality 2012, p. 7 (2012). https://doi.org/10.1145/2184305.2184308. http://dl.acm.org/citation.cfm?id=2184305.2184308
16. Lipka, N., Stein, B.: Identifying featured articles in Wikipedia: writing style matters. In: Proceedings of the 19th International Conference on World Wide Web 2010, pp. 1147–1148 (2010). https://doi.org/10.1145/1772690.1772847
17. Suzuki, Y., Nakamura, S.: Assessing the quality of Wikipedia editors through crowdsourcing. In: Proceedings of the 25th International Conference Companion on World Wide Web, WWW 2016 Companion, pp. 1001–1006. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva (2016). https://doi.org/10.1145/2872518.2891113
18. M.L.G. at the University of Waikato: Weka 3: Data mining software in Java. http://www.cs.waikato.ac.nz/ml/weka/
19. Warncke-wang, M., Cosley, D., Riedl, J.: Tell me more: an actionable quality model for Wikipedia. WikiSym **2013**, 1–10 (2013). https://doi.org/10.1145/2491055.2491063
20. Węcel, K., Lewoniewski, W.: Modelling the quality of attributes in Wikipedia infoboxes. In: Abramowicz, W. (ed.) BIS 2015. LNBIP, vol. 228, pp. 308–320. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-26762-3_27
21. M.W. Wikimedia: List of wikipedias. https://meta.wikimedia.org/wiki/List_of_Wikipedias
22. Wikipedia: Pomoc: jak napisać dobrą definicję. https://pl.wikipedia.org/?curid=757430

23. Wikipedia: Pomoc: styl - poradnik dla autorów. https://pl.wikipedia.org/?curid=3325631
24. Wikipedia: Wikipedia: manual of style/lead section. https://en.wikipedia.org/?curid=526968
25. WikiRank: Quality and popularity assessment of Wikipedia. http://www.wikirank.net
26. Wikisłownik: Indeks: polski - najpopularniejsze słowa 1–2000. https://pl.wiktionary.org/?curid=356429
27. Wilkinson, D.M., Huberman, B.a.: Cooperation and quality in Wikipedia. In: Proceedings of the 2007 International Symposium on Wikis WikiSym 2007, pp. 157–164 (2007). https://doi.org/10.1145/1296951.1296968. http://portal.acm.org/citation.cfm?doid=1296951.1296968
28. Wolinski, M., Milkowski, M., Ogrodniczuk, M., Przepiórkowski, A.: PoliMorf: a (not so) new open morphological dictionary for polish. In: LREC, pp. 860–864 (2012)
29. Woliński, M.: System znaczników morfosyntaktycznych w korpusie ipi pan. In: Polonica XXII-XXIII, pp. 39–55 (2003)
30. Wu, G., Harrigan, M., Cunningham, P.: Characterizing Wikipedia pages using edit network motif profiles. In: Proceedings of the 3rd International Workshop on Search and Mining User-Generated Contents, SMUC 2011, pp. 45–52. ACM, New York (2011). https://doi.org/10.1145/2065023.2065036
31. Xu, Y., Luo, T.: Measuring article quality in Wikipedia: Lexical clue model. In: 2011 3rd Symposium on Web Society (SWS), pp. 141–146. IEEE (2011). https://doi.org/10.1109/SWS.2011.6101286

NLP in OTF Computing: Current Approaches and Open Challenges

Frederik S. Bäumer(✉) and Michaela Geierhos

Paderborn University, Paderborn, Germany
{fbaeumer,geierhos}@mail.upb.de

Abstract. On-The-Fly Computing is the vision of covering software needs of end users by fully-automatic compositions of existing software services. End users will receive so-called service compositions tailored to their very individual needs, based on natural language software descriptions. This everyday language may contain inaccuracies and incompleteness, which are well-known challenges in requirements engineering. In addition to existing approaches that try to automatically identify and correct these deficits, there are also new trends to involve users more in the elaboration and refinement process. In this paper, we present the relevant state of the art in the field of automated detection and compensation of multiple inaccuracies in natural language service descriptions and name open challenges needed to be tackled in NL-based software service composition.

Keywords: Inaccuracy detection · Natural language software requirements

1 Introduction

Software requirements are challenging from a user perspective because they often allow a high degree of freedom in project implementation due to inaccuracies. In addition, the requirements are formulated in natural language (NL) and are therefore often ambiguous and partially incomplete. In the vision of On-The-Fly Computing[1], individual software requirements of end users have to be considered by the automatic composition of individual software services. Automated procedures are used to compensate shortcomings in the texts as well as possible but have noteworthy weaknesses: For example, it is very difficult to compensate missing non-standard information in case of incompleteness. Of course, it is possible to use default values or historical values here [14]. However, this is not always expedient and, above all, not always what the user wanted. Since users expect fast response times as well as on-the-fly results and do not want to deal with time-consuming compensation procedures, processing times are crucial in OTF Computing. Existing approaches for the improvement of NL requirement descriptions were often developed under different conditions (e.g. lower automation demand). Furthermore, existing requirement elaboration and refinement approaches are directed by experts, who do not necessarily consider the end-user perspective. In this paper, we discuss special challenges of service descriptions

[1] See https://sfb901.upb.de for more information about OTF Computing.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 559–570, 2018.
https://doi.org/10.1007/978-3-319-99972-2_46

processing in OTF Computing and their implications on Natural Language Processing (NLP) in order to suggest a road map for further research in this area (Sect. 2). We therefore provide an overview of the related work in Sect. 3 before we enter into a comprehensive discussion in Sect. 4. Finally, we conclude and give an outlook on future work in Sect. 5.

2 Natural Language Software Descriptions in OTF Computing

NLP in OTF computing is indispensable because service descriptions are the only source of information and therefore, as much information as possible must be extracted in the best possible quality. Service descriptions are a sub-type of requirement specifications, which are particularly characterized by their informal character (i.e. everyday language) and by the expected inaccuracy and incompleteness of natural language expressions. Thus, their degree of elaborateness depends on various individual preconditions (e.g. prior knowledge of the end users). For this reason, service descriptions can be seen as an informal document type, which is commonly called user-generated content [25]. If this topic is considered in the context of OTF Computing, the focus of existing work is on the application and development of semi-formal or formal specification languages for software services. First and foremost, this includes the software specification language (SSL), which was explicitly developed for the comprehensive specification of services and covers functional requirements as well as non-functional requirements [29]. However, as noted by Geierhos et al. [14, 15], any form of (semi)formal specification is unsuitable for end users. End users do not have the necessary expert knowledge, do not know certain information (e.g. (pre)conditions of a service and therefore cannot formally specify them [11]). In comparison to this, service descriptions can be written by end users, but are also unstructured, often incorrect in grammar and spelling, incomplete and ambiguous. This is still in contrast to the existing formal specification possibilities, which are available in OTF Computing up to now. For this reason, we want to address the following three issues in this paper.

2.1 Extraction of Canonical Core Functionalities

NL service descriptions of non-expert users often contain off-topic information and are little structured [15]. It is therefore very challenging to separate important from unimportant information [9]. In addition, it is challenging to extract the canonical core functionalities that are not often specified in any logical order. In this context, functional requirements are often understood as semantic categories (e.g. role, action, object) [9]. Furthermore, for the software service selection process (i.e. the decision which application will be included in a composition), the dependencies and other relationships between the requirements are important. Because of that it is necessary to extract the desired functionalities (process words) from all sentences and to put them into a functional sequence. An example is the sentence "I want to *send* e-mails to my friends: First I need to *write* them and then I want to *attach* my files". As shown in

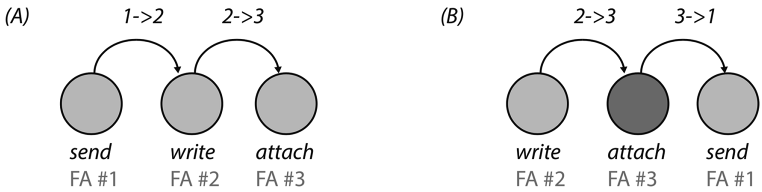


Fig. 1. Comparison of generated functional sequences [3]

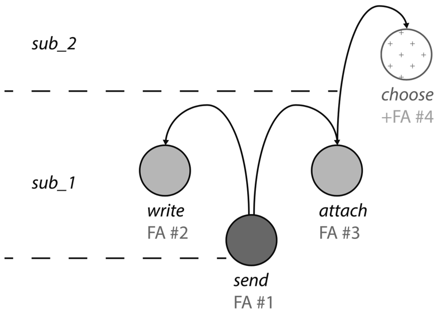


Fig. 2. Hierarchical arrangement of process words [3]

Fig. 1, existing information extraction techniques put software functions in a temporal or even causal order, which might have been intended by the end user.

If the order of execution is also considered, it is noticeable that the resulting functional sequence in Fig. 1(A) is not executable because the sending ("*send"*) an e-mail cannot possibly take place before writing ("*write"*) it. Consequently, the process words are at least dependent on the temporal arrangements (cf. Fig. 1B). However, they are not only subject to a temporal, but also to a hierarchical arrangement (cf. Fig. 2): Thus, sending is to be understood as a process word at first, but it also requires (in this example) the writing of a text and the attachment of a file ("*attach"*) as preconditions.

In addition to the functional sequence mentioned above, the subsequent expansion of existing concepts is needed. "*Attach*", for example, also presupposes an additional process step before – the selection ("*choose"*) of corresponding files. This process word did not literally appear in the initial service description (cf. Fig. 2). For this reason, an appropriate knowledge base is required, which contains process words, their hierarchical (semantic) relationships to other process words as well as dependencies, cognitive relations (e.g. synonymy) and context information (e.g. domain). Such a linguistic resource does not currently exist. Moreover, according to our knowledge, there is no processing method that extracts the inter-relationships between requirements as detailed as needed in our scenario.

2.2 Automatic Detection and Compensation of Inaccuracy

A large number of publications deals with ambiguity and other inaccuracies in requirement descriptions (e.g. [15, 19, 28, 33]). However, the service descriptions considered in OTF Computing are predestined for inaccuracies, due to the use of NL and little or no prior (expert) knowledge of the end user. Thus, it happens that software is developed according to given requirement descriptions but does not meet the intended requirements of the stakeholder [21] or even contains errors [12]. In particular, in the case of OTF Computing, which provides an automated composition of services, inaccuracies can lead to poor software [15]. In general, inaccuracies can be detected manually – for example with the help of word lists or checklists [20], supported by software or fully automated [3]. Therefore, there is a variety of methods that are difficult to compare [2, 30]. We distinguish between generalized approaches that can detect a multitude of inaccuracies and even more specialized ones that recognize specific linguistic characteristics. In both cases, linguistic resources are needed that cover the characteristics of the corresponding inaccuracy. Generally speaking, there are very few linguistic resources in the area of requirements engineering [31] and there is even less in the area of OTF Computing [4]. Another relevant point is the overall software performance, which is important for OTF Computing and which often has not the highest priority in software development in the field of NLP. Software solutions, that provide the detection and compensation of inaccuracies, should not be proprietary and should not dependent on external resources. An example therefor is the lexical disambiguation tool Babelfy [26, 27], which works very well in the OTF scenario [5] but is hosted on external servers. Network downtimes and unpredictable response times make integration into the OTF scenario rather difficult [3]. Moreover, third-party software integration in OTF Computing often fails because application programming interfaces are not provided, which are essential to reach an extensive automation.

2.3 Explainable Results

As shown before, it is already possible to automatically detect and compensate a large number of inaccuracies in service descriptions. But especially when a considerable large number of corrections has been made, it is important to inform the user about these corrections so that he or she can understand why the resulting software composition has certain characteristics. Although there are already approaches that visualize the corrections made or present comparisons by before/after (e.g. [5]), there is no explanatory text on the changes made. Here, the challenge arises from the fact that the resulting software service composition is strongly based on the initial input of the end users but is also characterized by the type of composition (the actual software service selection step) and by the available software services. This means that the desired transparency is not only reached by explaining how the detection and compensation of NL inaccuracies was done but must be a prerequisite for a better understanding.

3 Current State of Research

In the following, we present the state of the art for the extraction of canonical core functionalities and the automatic detection and compensation of inaccuracy.

3.1 Requirement Extraction

At the moment, there are only a few papers of different quality that are dedicated to the extraction of requirements from text. For example, Vlas and Robinson [34] perform an examination of unstructured and informal requirement descriptions in the area of Open Source software to learn more about freely formulated requirement texts. In addition, a tool called REaCT is available [8, 9], which uses machine learning (ML) procedures to detect on-topic statements in requirement descriptions and to transfer the essential components of functional requirements into a defined template. Dollmann [8] divides the requirement descriptions into individual sentences and transfers them to the classification component, which can subdivide sentences into off-topic and on-topic. If a classified sentence is on-topic and contains functional requirements, the extraction of attribute-value pairs is performed with the aim of iteratively filling the given template: The most important elements of the template are components (i.e. subject), actions (i.e. predicate), and objects (i.e. in-/direct object). Although REaCT convinces with good recall and precision, the approach fails due to the lack of suitable resources [4]. In this case, descriptions of requirements in NL are required, which are not (or hardly) publicly available [31]. For this reason, there are approaches that try to extract requirements without ML and are instead based on rules [7]. Due to the expected poor text quality of the service descriptions, however, there is still a lot of work to be done.

In addition to the above-mentioned contributions, there are mainly approaches that focus on the analysis and compensation of high-quality requirement descriptions or make far-reaching assumptions on text quality. They are therefore unsuitable for the application in the OTF Computing scenario.

3.2 Multiple Inaccuracy Detection and Compensation

The idea of combining several methods of inaccuracy detection and/or compensation can be found several times in the literature. Thus, there are mixed approaches that can detect different forms of ambiguity (e.g. [1, 32]) as well as methods that detect ambiguity and incompleteness (e.g. [10, 17, 22]). An overview on existing disambiguation techniques in the context of NL requirements is given by Husain and Beg [18] and Shah and Jinwala [30]. Shah and Jinwala differentiate approaches with regard to the degree of automation, chosen methods (rule-based, ontology-based, etc.) and technologies used (e.g. Stanford Parser) [30]. Another extensive overview of existing approaches is given by Bano [2], focusing on empirical work. In the following, hybrid approaches of ambiguity/incompleteness detection and compensation are listed concerning the dimensions “Defects”, “Objectives”, “Input/Output” (I/O) and “Interaction”. “Defects” reflects the coverage of the approaches. NL2OCL and SR-Elicitor are tools that reach a low coverage while QuARS [24] and QuARS$_{\text{express}}$ [6] cover a variety of inaccuracies. In addition, the methods also differ in the objectives. While QuARS pursues the goal of detecting as many deficits as possible in requirement descriptions, NL2OCL [1] and SR-Elicitor [33] are aimed at detection and compensation – both without user interaction. In contrast to this, RESI (Requirements Engineering Specification Improver) requires a high user interaction in the compensation of inaccuracies. In the following, NLARE (Natural Language Automatic Requirement Evaluator) [16, 17], RESI [23] and CORDULA (Compensation of Requirements Descriptions Using Linguistic Analysis) [5] are presented in more detail to highlight the differences (Fig. 3).

NLARE is a mixed approach concentrating on functional requirements and the detection of ambiguity, incompleteness and atomicity [16]. The authors see ambiguity in the occurrence of adjectives and adverbs that can be in- or decreased[2]. Incompleteness refers to the comparison with given "W-questions" ("*Who*", "*What*", "*Where*", "*When*"), and atomicity describes a quality criterion that a single sentence should only contain a single requirement. NLARE uses NLTK[3] and regular expressions to process NL. The sequential processing pipeline includes sentence boundary detection and tokenization as well as spelling correction. As an output, users get hints such as "The requirement is ambiguous because it contains the word 'earlier' and 'later'" [16]. There is no compensation or further assistance. But RESI [22] is different from NLARE – especially in terms of user interaction, flexibility and coverage of linguistic defects. RESI is able to load requirement specifications as a graph and automatically check them for linguistic inaccuracies. If inaccuracies are found, RESI initiates a user dialog [23]. This dialog does not only refer to the problematic text passages, but also gives explicit compensatory hints for various kinds of inaccuracies (e.g. incomplete process words). Therefore, the integration of resources is required. This way, it is possible to detect the inaccuracies (rules) on the one hand and to provide additional information for the compensation on the other hand (i.e. different ontologies). Another tool, similar to RESI, is CORDULA [5]. The CORDULA system detects and compensates so-called language deficiencies (e.g., ambiguity, vagueness and incompleteness) in requirements written by non-expert end users. Since CORDULA is developed in the OTF context, it supports the search for suitable software services that can be combined in a composition by transferring requirement specifications into canonical core functionalities. Using linguistic indicators makes it possible to optimize the individual text quality in a data-driven and needs-oriented manner by deviating from the classical text analysis pipeline: Its distinguishing feature is the *ad hoc* configuration of the compensation pipeline, triggered by the deficiencies detected in the service specifications of end users. A disadvantage is a slow execution time [3]. This means, for example, that no compensation can be made when contradictory information from different compensation mechanisms is provided (stalemate situation) or information needed for processing is simply missing [5]. As can be seen here, there are quite a few tools that can detect several types of inaccuracies in NL requirement descriptions and even partially compensate them. However, there is still no approach that meets all the prerequisites for processing NL service descriptions in OTF Computing.

---

[2] This understanding of ambiguity is often summarized in literature as "vagueness".

[3] Visit https://www.nltk.org for more information.

		NLARE	NLARE2	SREE	**RESI**	**QuARS**	NL2OCL	SR-Elicitor	SRR-Director	$QuARS_{express}$	Smella	AQUSA	CORD
Defects	Lexical Ambiguity	●	●	●	●	●	●	●	●	●	●	●	●
	Syntactic Ambiguity	○	○	●	○	●	●	●	●	●	○	○	●
	Referential Ambiguity	○	○	●	○	●	○	○	○	●	○	○	●
	Vagueness	○	○	●	●	●	○	○	●	●	●	○	●
	Incompleteness	●	●	●	●	●	○	○	○	●	●	●	●
	Readability	○	○	○	○	●	○	○	○	●	○	●	○
	Consistency	○	○	○	○	●	○	○	○	●	●	●	○
	Isolation	●	●	○	○	●	○	○	○	●	○	●	○
Obj.	Detection	●	●	●	●	●	●	●	●	●	●	●	●
	Compensation	○	○	○	●	○	●	●	○	○	○	○	●
I/O	Structured input	○	○	○	●	○	●	○	○	○	●	●	●
	Structured output	○	○	○	●	○	●	●	○	○	○	○	●
Interact.	High	○	○	○	●	○	○	○	○	○	○	○	○
	Medium	○	○	○	○	●	○	○	○	●	○	○	○
	Low	●	●	●	○	○	○	○	●	○	○	○	○
	None	○	○	○	○	○	●	●	○	○	●	●	●

Fig. 3. Detection and compensation approaches [3]

4 Open Challenges: A Discussion

NLP pipelines in requirements engineering have been an issue of substantial research for years. Here, the task of NLP is to analyze and enhance the original service descriptions of end users without distorting the actual intention. However, inaccuracies should be corrected without frightening end users with technical or linguistic details – a challenging task. Such an NL processing pipeline which creates structured compositional templates from unstructured service descriptions has already been designed and implemented as a prototype [3, 5]. And yet many questions have remained unanswered and not all sub-goals (such as high performance) have been achieved.

Existing methods for compensating inaccuracies are designed as special applications that focus only on a specific linguistic phenomenon. However, we need much more approaches that pursue an overall strategy and also keep track of the individual deficits in the entire requirement description. For example, how does the compensation of incompleteness affect possible ambiguities? In addition, it is rarely questioned whether the application of a compensation step is really necessary to understand an affected requirement – does the ambiguity concern, for example, essential elements of a requirement, or can the compensation be skipped in terms of performance? Another point that is rarely covered by existing procedures is the synergy between individual compensation components. For instance, do information that has been gained during compensating incompleteness help to speed up syntactic disambiguation or *vice versa*? However, this presupposes that NLP pipelines are no longer regarded as strictly sequential but that the individual components interact more effectively. It is not enough to simply extract requirements from the input texts and provide them in a structured way for the next processing step. End users will never think of all the requirements that need to be specified in order to develop software. In addition, end users will always describe at different levels, mostly at the highest (front-end) level ("I want to send e-mails") and not at the component (back-end) level. Here, information extraction and compensation must work together to identify hierarchical dependencies between process words and reconstruct missing links.

As already mentioned, there is a lack of resources covering domain-specific properties. However, there are extensive linguistic resources such as BabelNet [26, 27] that can be used as starting point. In addition to the existing linguistic information, it is important to refine requirement-specific information like hierarchical relationships between some process words (cf. Fig. 4). Moreover, there is the aforementioned lack of linguistic resources that reflect the characteristics of the texts expected in OTF Computing [4, 31]. This includes resources for compensation as well as real service descriptions. This is a challenge, because we only get NL service descriptions from end users when the system exists, but this will exist only when real texts are available e.g. for training. There are approaches that use related texts [9], but this is only a simulation of service descriptions: Some features can be different; some are not covered at all.

Up to now, the aim has been to achieve maximum automation, leaving users out of the loop as far as possible. Moreover, user participation in the requirement elaboration process is time-consuming. But the users know best what they actually expect from the software, so that user interaction seems reasonable. Here, approaches must be developed that make the efficient integration of end users possible. If information is missing or ambiguous, users can quickly make decisions if the system supports them. For example, additional user support could be provided by a chat bot (cf. Fig. 5), which integrates end users into the requirements elaboration and refinement process and gives

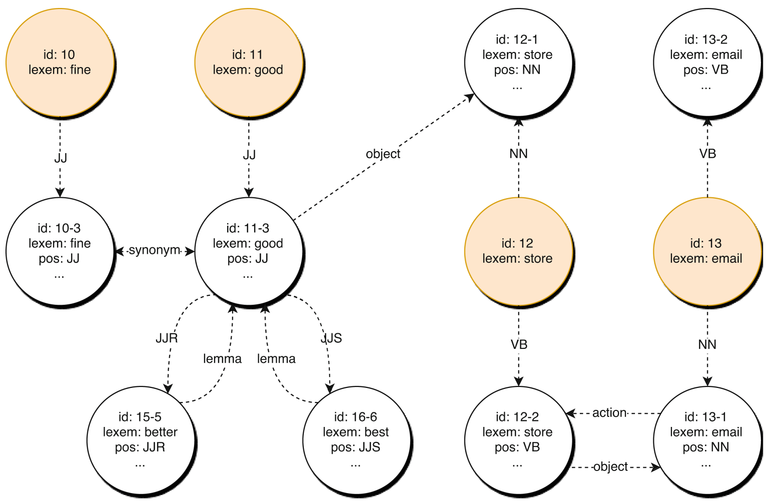


Fig. 4. Exemplary NL requirement knowledge base

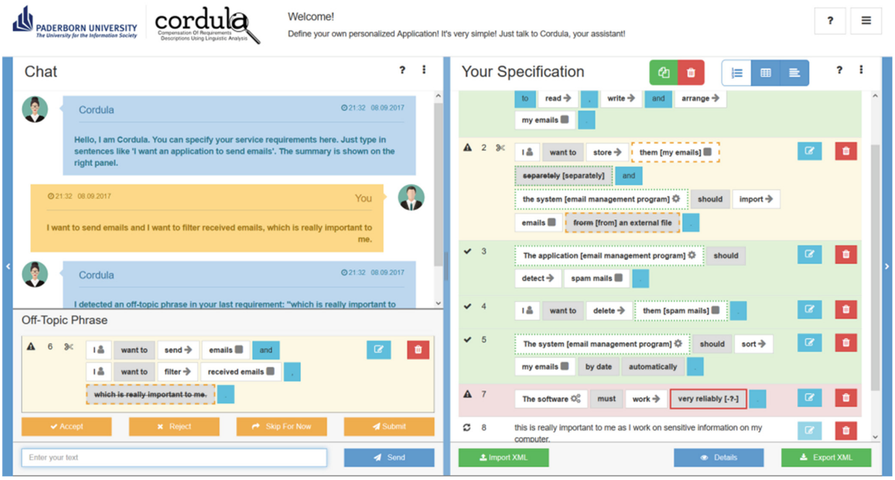


Fig. 5. NL software requirement compensation via chat bot [13]

them hints (e.g. examples) depending on the situation [13]. The chat bot can highlight inaccuracies and can offer tailored action proposals. While chatting, the end users can either confirm the system's suggestion or reject it. This means that users communicate with the system in two ways: On the one hand, users express their requirements via the chat interface and receive an immediate response. On the other hand, they will be asked to react to certain circumstances (e.g. missing information) and the chat bot adapts to the situation. As shown on the right-hand side of Fig. 5, end users are informed what happens with their initial input text and are able to edit the resulting software requirements [13]. Chat bots can thus help within the compensation process and illustrate the results to end users.

5 Conclusion

The challenges mentioned in this paper can be also found in traditional requirements engineering. Because of the specific nature of OTF Computing, NLP approaches developed so far do not achieve the required execution times here. Therefore, there is a lot to be done: Attention still needs to be paid to the development of methods for extracting requirements as well as to the detection and compensation of inaccuracies. This raises other issues such as the lack of resources but also the lack of interoperability of individual compensation components, ways of efficiently involving end users without overburdening them, and much more. Through the discussion (Sect. 4), we hope to have presented the open questions and current challenges for the NLP in the context of OTF Computing and hope to have given some structure to the following research in this area. At the same time, we regard these shortcomings and challenges as a road map for our own research in the field of NLP pipeline configuration and execution. In particular, we will use modern chat technology to conduct targeted communication with end users as part of the compensation steps. Through this procedure, missing information can be requested, and the user is supported in answering the questions with examples etc. Furthermore, the results generated during the compensation process can be explained during the offered chat, which increases the usability for end users.

Acknowledgements. This work was partially supported by the German Research Foundation (DFG) within the Collaborative Research Center "On-The-Fly Computing" (SFB 901). Furthermore, we thank our student assistant Edwin Friesen for his contribution.

References

1. Bajwa, I.S., Lee, M., Bordbar, B.: Resolving syntactic ambiguities in natural language specification of constraints. In: Gelbukh, A. (ed.) CICLing 2012. LNCS, vol. 7181, pp. 178–187. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-28604-9_15
2. Bano, M.: Addressing the challenges of requirements ambiguity: a review of empirical literature. In: Proceedings of the 5th International Workshop on EmpiRE, pp. 21–24. IEEE, Ottawa, August 2015
3. Bäumer, F.S.: Indikatorbasierte Erkennung und Kompensation von ungenauen und unvollständig beschriebenen Softwareanforderungen. Ph.D. thesis, University of Paderborn, Paderborn, Germany, July 2017. ISBN: 978-3-942647-91-5
4. Bäumer, F.S., Dollmann, M., Geierhos, M.: Studying software descriptions in SourceForge and app stores for a better understanding of real-life requirements. In: Sarro, F., Shihab, E., Nagappan, M., Platenius, M.C., Kaimann, D. (eds.) Proceedings of the 2nd ACM SIGSOFT International Workshop on App Market Analytics, pp. 19–25. ACM, Paderborn, September 2017
5. Bäumer, F.S., Geierhos, M.: Flexible ambiguity resolution and incompleteness detection in requirements descriptions via an indicator-based configuration of text analysis pipelines. In: Proceedings of the 51st Hawaii International Conference on System Sciences, Big Island, Waikoloa Village, USA, pp. 5746–5755 (2018)
6. Bucchiarone, A., Gnesi, S., Fantechi, A., Trentanni, G.: An experience in using a tool for evaluating a large set of natural language requirements. In: Proceedings ACM Symposium on Applied Computing, SAC 2010, pp. 281–286. ACM, New York (2010)
7. Caron, M., Bäumer, F.S., Geierhos, M.: Back to basics: extracting software requirements with a syntactic approach. In: Schmid, K., et al. (eds.) Joint Proceedings of REFSQ-2018 Workshops, Doctoral Symposium, Live Studies Track, and Poster Track Co-located with the 23rd International Conference on Requirements Engineering: Foundation for Software Quality (REFSQ 2018). CEUR-WS.org (2018)
8. Dollmann. M.: Frag die Anwender: Extraktion und Klassifikation von funktionalen Anforderungen aus User-Generated-Content. Master thesis, University of Paderborn, Paderborn, Germany, March 2016
9. Dollmann, M., Geierhos, M.: On- and off-topic classification and semantic annotation of user-generated software requirements. In: Proceedings of the Conference on EMNLP. ACL, Austin, November 2016
10. Fabbrini, F., Fusani, M., Gnesi, S., Lami, G.: The linguistic approach to the natural language requirements quality: benefit of the use of an automatic tool. In: Proceedings of the 26th Annual NASA Goddard Software Engineering Workshop, pp. 97–105. IEEE, Greenbelt, November 2001

11. Ferrari, A., dell'Orletta, F., Spagnolo, G.O., Gnesi, S.: Measuring and improving the completeness of natural language requirements. In: Salinesi, C., van de Weerd, I. (eds.) REFSQ 2014. LNCS, vol. 8396, pp. 23–38. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-05843-6_3
12. Firesmith, D.: Common requirements problems, their negative consequences, and the industry best practices to help solve them. J. Object Technol. **6**(1), 17–33 (2007)
13. Friesen, E., Bäumer, F.S., Geierhos, M.: CORDULA: software requirements extraction utilizing chatbot as communication interface. In: Schmid, K., et al. (eds.) Joint Proceedings of REFSQ-2018 Workshops, Doctoral Symposium, Live Studies Track, and Poster Track Co-located with the 23rd International Conference on Requirements Engineering: Foundation for Software Quality (REFSQ 2018). CEUR-WS.org (2018)
14. Geierhos, M., Bäumer, F.S.: How to complete customer requirements: using concept expansion for requirement refinement. In: Métais, E., Meziane, F., Saraee, M., Sugumaran, V., Vadera, S. (eds.) NLDB 2016. LNCS, vol. 9612, pp. 37–47. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-41754-7_4
15. Geierhos, M., Schulze, S., Bäumer, F.S.: What did you mean? Facing the challenges of user-generated software requirements. In: Loiseau, S., Filipe, J., Duval, B., van den Herik, J. (eds.) Proceedings of the 7th ICAART, Special Session on PUaNLP 2015, pp. 277–283. SCITEPRESS – Science and Technology Publications, Lissabon (2015)
16. Huertas, C., Juárez-Ramírez, R.: NLARE, a natural language processing tool for automatic requirements evaluation. In: Proceedings of the CUBE International Information Technology Conference, CUBE 2012, pp. 371–378. ACM, New York (2012)
17. Huertas, C., Juárez-Ramírez, R.: Towards assessing the quality of functional requirements using English/Spanish controlled languages and context free grammar. In: Proceedings of the 3rd DICTAP, pp. 234–241. SDIWC, Ostrava, July 2013
18. Husain, S., Beg, R.: Advances in ambiguity less NL SRS: a review. In: Proceedings of ICETECH 2015, pp. 221–225. IEEE, Coimbatore, March 2015
19. Kamsties, E.: Understanding ambiguity in requirements engineering. In: Aurum, A., Wohlin, C. (eds.) Engineering and Managing Software Requirements, pp. 245–266. Springer, Heidelberg (2005). https://doi.org/10.1007/3-540-28244-0_11
20. Kamsties, E., Berry, D.M., Paech, B.: Detecting ambiguities in requirements documents using inspections. In: Proceedings of 1st WISE, Paris, France, pp. 68–80 (2001)
21. Kamsties, E., Paech, B.: Taming ambiguity in natural language requirements. In: Proceedings of the 13th ICSOFT-EA, Paris, France, pp. 1–8, December 2000
22. Körner, S.J.: RECAA - Werkzeugunterstützung in der Anforderungserhebung. Ph.D. thesis, Karlsruher Institut für Technologie. KIT Scientific Publishing, Karlsruhe, February 2014
23. Körner, S.J., Brumm, T.: Natural language specification improvement with ontologies. Int. J. Semant. Comput. **03**(04), 445–470 (2010)
24. Lami, G.: QuARS: a tool for analyzing requirements. Technical report ESC- TR-2005-014, Carnegie Mellon University, September 2005
25. Moens, M.-F., Li, J., Chua, T.-S.: Mining User Generated Content. CRC Press, Leuven/Beijing/Singapore (2014)
26. Navigli, R., Ponzetto, S.P.: BabelNet: the automatic construction, evaluation and application of a wide-coverage multilingual semantic network. In: Artificial Intelligence, vol. 193, pp. 217–250. Elsevier, Essex, December 2012
27. Navigli, R., Ponzetto, S.P.: Joining forces pays off: multilingual joint word sense disambiguation. In: Proceedings of the 2012 Joint Conference on EMNLP and CONLL, pp. 1399–1410. ACL, Jeju (2012)

28. Pekar, V., Felderer, M., Breu, R.: Improvement methods for software requirement specifications: a mapping study. In: Proceedings of the 9th QUATIC, pp. 242–245. IEEE, Guimarães, September 2014
29. Platenius, M.C. et al.: An Overview of Service Specification Language and Matching in On-The-Fly Computing (v0.3). Technical report Tr-ri-16-349, Software Engineering Group, Heinz Nixdorf Institut, University of Paderborn, Germany, January 2016
30. Shah, U.S., Jinwala, D.C.: Resolving ambiguities in natural language software requirements: a comprehensive survey. SIGSOFT **40**(5), 1–7 (2015)
31. Tichy, W.F., Landhäußer, M., Körner, S.J.: nlrpBENCH: a benchmark for natural language requirements processing. In: MK Software Engineering and Management (2015)
32. Tjong, S.F., Berry, D.M.: The design of SREE – a prototype potential ambiguity finder for requirements specifications and lessons learned. In: Doerr, J., Opdahl, A.L. (eds.) REFSQ 2013. LNCS, vol. 7830, pp. 80–95. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-37422-7_6
33. Umber, A., Bajwa, I.S.: Minimizing ambiguity in natural language software requirements specification. In: Proceedings of the 6th ICDIM, pp. 102–107. IEEE, Melbourn, September 2011
34. Vlas, R., Robinson, W.N.: A rule-based natural language technique for requirements discovery and classification in open-source software development projects. In: Proceedings of the 44th HICSS, pp. 1–10. IEEE, Kauai (2011)

Text Augmentation Techniques for Document Vector Generation from Russian News Articles

Christoffer Aminoff[1], Aleksei Romanenko[1,2], Onni Kosomaa[1], and Jouko Vankka[1(✉)]

[1] National Defence University, 00861 Helsinki, Finland
jouko.vankka@mil.fi
[2] Lappeenranta University of Technology, 53850 Lappenranta, Finland

Abstract. In this paper, a document classification system is enhanced through the construction of a text augmentation technique by testing various Part-of-Speech filters and word vector weighting methods with nine different models for document representation. Subject/object tagging is introduced as a new form of text augmentation, along with a novel classification system grounded in a word weighting method based on the distribution of words among classes of documents. When an augmentation including subject/object tagging, a nouns+adjectives filter and Inverse Document Frequency word weighting was applied, an average increase in classification accuracy of 4.1% points was observed.

Keywords: Natural language processing · Classification algorithms
Data preprocessing · Text processing

1 Introduction

The performance of a document classification system can be improved with a number of text augmentation techniques, including text preprocessing and word weighting. Stop-word and Part-of-Speech (POS) filtering may be used to filter out tokens that are seen to carry little to no semantic content, or those that indeed distort the semantics of the document as a whole.

The aim of this piece of research is to analyse the impact of various text augmentation techniques, including Part-of-Speech filtering, subject/object tagging and word weighting, on the classification performance of a variety of vector representation models on Russian-language text. Finally, the paper proposes and measures the effectiveness of a novel word weighting and Subject-Object (SO) tagging approach that is based on class-by-class frequency statistics for words.

2 Related Work

2.1 Text Preprocessing

Two common forms of preprocessing that are applied to text before computing representations for it are lemmatisation and POS. Lemmatisation normalises

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 571–586, 2018.
https://doi.org/10.1007/978-3-319-99972-2_47

each token in the corpus so that it is replaced with its base form. This collapses different morphological forms of the same word into one and reduces the number of unique tokens in the corpus, thereby, potentially reducing the computational cost of further processing of the corpus. Thus, lemmatisation is most useful for morphologically rich languages, including Slavic languages like Russian [9], as well as Uralic languages like Finnish [25]. This can in some cases, however, lead to the loss of information, and thereby, reduced semantic accuracy [24].

POS filtering is one of many techniques that enable the selection of only certain types of words for use in the computation of document representations, with other such techniques including stop-word removal and lexical filters [25]. The aim is to mitigate the impact of words irrelevant to the semantics of the sentence or document as a whole. It should be noted that increased accuracy is only attained with appropriate POS filters, as has been found in [6]. In order to implement POS filtering, one needs to either have a POS-annotated corpus or software that can POS tag sentences appropriately. Such POS taggers exist for specific languages, such as MyStem for Russian [21], as well as for interlingual use, such as the RDRPOSTagger [16].

As POS tagging only provides information about individual words, the method lacks context information. In order to correct this flaw, several studies have been undertaken. The general idea is to analyse or train a classifier that is capable of producing semantic tags that represent the role of an individual word in the sentence. In [8], local tags were identified for the brand, type, quantity and descriptive words for a set of food descriptions, while [2] proposes a generalised semantic classification task into multiple role-based tags.

2.2 Word Weighting

Instead of simple averaging, vectors of words within a document can be weighted before being combined in order to produce enhanced document vectors, an instance of which shall be called Weighted Document Vector (WDV). Here, one can take advantage of the typically uneven distribution of unique words between documents, as well as between classes of documents within a corpus. Inverse Document Frequency (IDF) [7] weighting [5] is an obvious option as it can be applied along the same lines as the Term Frequency - Inverse Document Frequency (TF-IDF) weighting of vector features in Bag of Words (BOW) models [20]. Another alternative is to use supervised weighting, where the documents in a corpus are split into a number of labelled classes and the word vectors are weighted according to the distribution among these classes of the words and/or the documents where the words are found. One such weighting method is Relevance Frequency (RF) [10], where the documents are classed as either positive or negative. Those words that appear more frequently in the positive category than the negative are given more weight. In document classification tasks, one can also weight words based directly on the classes used in the classification task, which is what Class Frequency (CF) [12] weighting does. The class-based weighting of word vector dimensions is another possible approach, tested in [22]. To the authors' best knowledge, RF and CF weightings have so far only been used for feature weighting in BOW-like document representations. We apply these to the weighting of word embeddings.

3 Methodology

3.1 Evaluation Overview

Figure 1 presents an overall flow chart of the preparation process performed on the input texts. The details on each step are presented in the following subsections of this section.

At the first stage, the corpus was lemmatised (Subsect. 3.2). Then, a POS filtering scheme was applied. The resulting subset of words was then analysed for subject/object tagging and appropriately extended. Then, a weighting statistic was computed. These are described in more detail in Subsect. 3.3. Finally, the word embedding model was trained, using the augmented dataset and a WDV was prepared for each document using precomputed word weights (Subsect. 3.4). The quality of the resulting document vectors was evaluated in a classification task (Subsect. 3.5).

3.2 Corpus

Each of the embeddings used was trained on the OpenCorpora [13] corpus in its form as of 4 September 2017. The corpus consists of Russian-language texts of various lengths, where around half of these are news articles. The rest include amongst other things blog posts, Wikipedia articles and judicial texts. The corpus was in an Extensible Markup Language (XML) format in its raw form, and many of the documents contained in the corpus had useful tags that facilitated the creation of classes. The corpus was preprocessed in one of two ways: In the initial iteration, all words were lemmatised with the Python implementation of MyStem [21]. In subsequent iterations, the words were "analysed", which included both the lemmatisation and POS tagging of words. Based on the POS tagging, the POS filter could then be implemented. In all iterations, those tokens that contained any non-alphanumeric characters except hyphens were filtered out.

3.3 Augmentation Techniques

Part-of-Speech Filtering. POS tagging was performed using the MyStem [21] analyser by Yandex. The analysis provides distinctive tags for nouns (N), verbs (V), adverbs (ADV), adjectives (A), pronouns and punctuation. The following combinations of tagged words were kept in different scenarios:

- N - nouns
- N+A - nouns and adjectives
- N+V - nouns and verbs
- N+ADV - nouns and adverbs

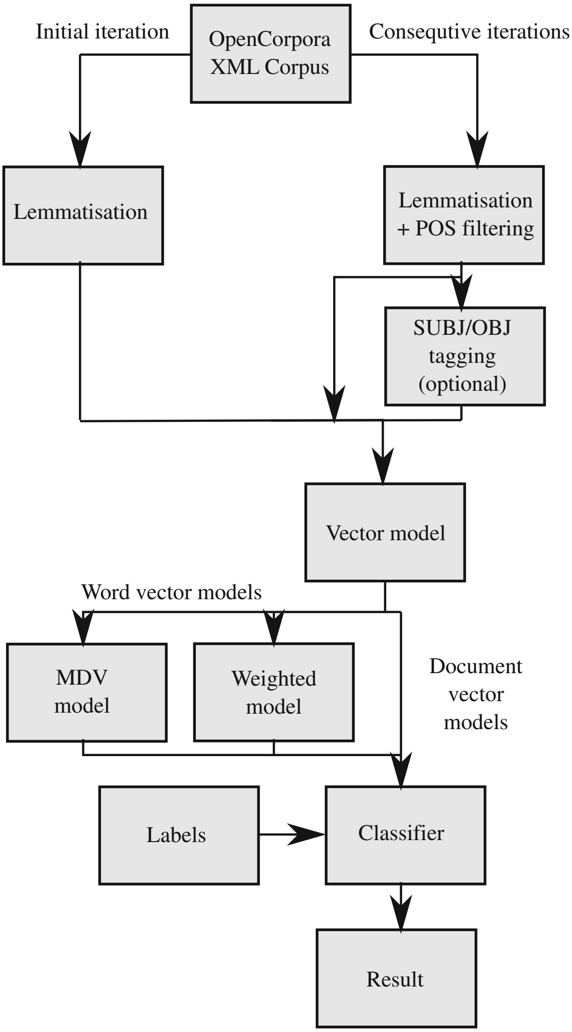


Fig. 1. Augmentation flow chart

– N+A+V - nouns, adjectives and verbs
– N+A+ADV - nouns, adjectives and adverbs
– N+V+ADV - nouns, verbs and adverbs
– N+A+V+ADV - nouns, adjectives, verbs and adverbs.

As per standard practice in recent studies [6,25], nouns were kept in all scenarios. Strings with at least three characters that couldn't be analysed, which were mostly strings with non-Cyrillic characters, as well as numbers were always included as they were assumed to carry document-specific information.

Subject-Object Tagging. In order to perform grammatical subject-object tagging, a basic analysis was performed for all words, although in practice, it only affected words tagged as nouns or adjectives. A word was considered to be a potential subject when it had a nominative case marked in the list of possible cases by MyStem. The word was considered to be a potential object when it had a non-nominative case marked in the list of possible cases. Finally, if a word had both nominative and non-nominative cases it was considered both a possible subject and object at the same time. This simplification was applied because the Russian language is known to have ambiguous cases where it is impossible even for a human to classify a word as either a subject or an object without knowing the global context or a priori knowledge of relations that are possible for any given subjects and objects. A good example of such an ambiguous sentence is: Дерево (Tree) подпирает (to support) столб (pole). Both nouns in this sentence have an ambiguous nominative-accusative case and only the word order weakly suggests that the tree supports the pole. The resulting tagged corpus contained the original tokens, followed by one or both possible tagged versions of the same token, or none if a grammatical case wasn't identified for the token. This was done to link semantically different forms of a word with each other and allow the selected embedding model to make distinctions between those forms.

Weighting. Word weighting was performed in two stages. First, the weights for the i-th word of the corpus were computed in accordance with the desired weighting equation, chosen out of Eqs. (3), (5) or (8). Then, weighted document vectors were computed using the weighted sum equation

$$W(J) = \frac{\sum\limits_{i \in J} v_i \cdot w_i}{M_W}, \tag{1}$$

where i is the word id, J is the set of words in a given document, v_i is the embedding vector for word i, w_i is weighting quotient and M_W is the normalisation quotient specific to the weighting method W, defined by Eqs. (2), (4) or (7). The normalisation quotient was chosen for each weighting method based on preliminary tests.

IDF. The IDF is a metric that allows one to lower the influence of words frequent across the corpus. The IDF for a word i was computed as follows:

$$IDF(i) = \log\left(\frac{|S|}{|\{s \in S : i \in s\}|}\right) \tag{2}$$

where S is the set of all sentences in the corpus.

The normalisation factor here transforms every document vector into a unit vector:

$$M_{IDF}(i) = \left|\sum_{i \in J} v_i \cdot w_i\right| \tag{3}$$

RF. RF [10] gives additional weight to words appearing more frequently than not in documents classed as "positive", i.e. documents deemed to be relevant. We define the positive category as the set of Chaskor news articles belonging to one the five topic categories used in classification. For a word i, the RF weight is computed with the following equation:

$$RF(i) = \log\left(2 + \frac{|\{s \in S_+ : i \in s\}|}{|\{s \in S_- : i \in s\}|}\right) \tag{4}$$

where S_+ and S_- are the sets of sentences appearing in documents belonging to the "positive" and "negative" class respectively. Here, the normalisation quotient is the sum of the weights of the words in the document:

$$M_{RF}(i) = \sum_{i \in J} w_i \tag{5}$$

Parallel IDF-CF. With CF [12], one can give additional weight to words prevalent in a specific class. The CF of a word varies depending on the class of the document it is situated in. In this paper, a sentence-based modification of the original weighting method was used. For instances where a word i appears in a sentence within a document belonging to a class c, the CF is computed as follows:

$$CF(i,c) = \frac{|\{s \in S_c : i \in s\}|}{|S_c|} \tag{6}$$

where S_c represents the set of all sentences appearing in documents belonging to the class c. The information contained in IDF and CF can be combined to form Inverse Document Frequency - Class Frequency (IDF-CF) weights:

$$IDF\text{-}CF(i,c) = \log\left(\frac{|S|}{|\{s \in S : i \in s\}|}\right) \cdot \frac{|\{s \in S_c : i \in s\}|}{|S_c|} \tag{7}$$

This gives the IDF-CF of a word i with respect to a class c.

Here as well, the normalisation quotient normalises a document vector with respect to the sum of its word weights:

$$M_{IDF-CF}(i) = \sum_{i \in J} w_i \tag{8}$$

However, since the classifier naturally does not know with certainty which class any given test document belongs to, there will be as many possible weights applicable to each word in the document as there are classes. In order to solve this problem, a *parallel IDF-CF* structure is proposed, where for each document, parallel vector representations are computed, one corresponding to each class. In other words, for each class, one computes a separate IDF-CF-weighted document vector assuming that the document belongs to that class, meaning that weights are computed for the tokens in the document with respect to the class. This weighting method therefore requires its own classification system.

Then, for each class-vector pair, the classifier, fitted with correctly-weighted document vectors, must predict the probability of the document belonging to the class when represented by the vector. The vector that gives the highest probability to its corresponding class is chosen as the sole representation for the document and is passed to the classifier for class prediction in the normal fashion.

3.4 Word Vector Embeddings

Six word embedding models were used: a Global Vectors (GloVe) [18] model, Swivel [23], as well as two word2vec and two fastText [3] models, one of each type trained using the skip-gram [15] algorithm, and the other using the continuous bag of words (CBOW) [14]. Two doc2vec paragraph vector models were also used, one DM and one DBOW model [11]. Finally, a BOW-TF-IDF model was computed. For the word2vec, fastText and doc2vec models, their implementations in the Gensim [19] library were used. This library also helped in the construction of BOW-TF-IDF vectors.

Each model was trained on the corpus as a whole, with documents used as inputs for the document vector models and sentences within the documents for the rest. The BOW-TF-IDF model, however, used Term Frequency (TF) data from full documents and IDF data based on token appearance in sentences, where each sentence is treated as a "document" for IDF purposes. This approach was chosen for two reasons: in order to limit the influence of longer documents so that one document can contribute more than once towards the IDF of a word, and because it was found to perform slightly better than purely document-based IDF.

For each word vector model, at least a mean document vector Mean Document Vector (MDV) was constructed for each document in the corpus. When necessary, this was supplemented by calculating various WDV. The doc2vec models already had document vectors right after training. "Default" values set in the Gensim [19] library were used for all hyperparameters in the training of all models apart from Swivel, where 100-dimensional embeddings were chosen to match most of the other models. This meant that all models except for GloVe had context windows of size 5 and 100-dimensional word vectors, whereas GloVe had these at 10 and 30 respectively. The deviating hyperparameters for GloVe were kept since tests where a window size of 5 and 100 dimensions were used recorded very poor results. The window size was doubled whenever subject/object tagging was included in the corpus. Further training hyperparameters were as follows for word2vec-based models: the hierarchical softmax was disabled, negative sampling was enabled with a sample size of 5, the initial learning rate was set at 0.025, and the minimum frequency for words at 5.

3.5 Evaluation of Performance in Document Classification

The evaluation was performed in three stages. At the first stage, the performance of the models was evaluated, with no text augmentation. Then, the best POS filtering, subject-object tagging scheme and word weighting method for WDV calculation were selected with respect to the four best-performing models: BOW-TF-IDF, Swivel, word2vec-skip-gram and doc2vec Distributed Bag of Words (DBOW). Finally, the selected combination of augmentation parameters was tested again on all models to analyse the impact it had on the performance of the classifiers.

3.6 Classification Tests

Scikit-learn [17] implementations of Support Vector Machine (SVM) were utilised as classifiers. Only Chastny Korrespondent articles were used directly in classification tests. This is because they have specific topics tagged in their XML originals that are listed on the OpenCorpora website. The topics also enabled the formation of sufficiently large classes. Of them, the five that had the largest number of texts belonging to them were chosen. These were 'Общество' (Society), 'В мире' (World events, lit. 'In the World'), 'Культура' (Culture), 'Медиа' (Media) and 'Технологии' ('Technologies').

A classification test was conducted as follows: First, out of the text documents that had one of the above as its topic, 750 were picked at random, 150 from each topic category. This set was then further divided, also at random, into 5 subsets, with each of them containing 150 documents, 30 from each topic category. Next, 5-fold validation was performed on the subsets. Here, the SVM classifier was first trained on 4 of the subsets, where the vectors of each document contained in these, paired with their corresponding topics as labels, were fed to the classifier. Then, having been trained, the classifier was tasked with predicting the topic of each document in the remaining subset, none of which it had seen previously. This step was repeated four more times so that each subset acted as the test set exactly once. The results of the five sub-tests were then combined to yield the final result, including the overall classification accuracy as well as a confusion matrix.

3.7 Choice of Optimal Classifiers

Because it was found that the accuracy of a classifier varies significantly depending on the dataset, different SVM classifiers with different configurations were tested in order to find an approximation of an optimal classifier for each document representation model. This was in practice done by repeating the classification test on different combinations of values for parameters relevant to classification performance, types of support vector classifiers (SVC vs. NuSVC vs. LinearSVC) and kernel functions. Only the result obtained with the best combination was recorded for each model. MDV vectors were used for word embedding models.

3.8 Evaluation of Augmentation Parameters

16 different POS-filtered versions of the corpus were created, two for each POS filter, one with and one without subject/object tagging. For each version, IDF,

RF and CF weights were computed for each unique token, with tagged tokens counting as separate from their untagged form.

The four best-performing models from the first iteration were chosen, namely BOW-TF-IDF, Swivel, word2vec-skip-gram and doc2vec-DBOW. Each of these was trained on each version of the corpus, creating 16 versions of each model. The size of the context window (where applicable) was doubled for those versions of the corpus that included subject/object tagging, accommodating tagged repetitions of the same word. 100 classification tests were run on each version of both BOW-TF-IDF and doc2vec-DBOW. For each version of Swivel and word2vec-skip-gram, an MDV model as well as 3 WDV models, one for each weighting method, were constructed and tested, also with runs of 100 tests. The average result was recorded for each set of 100 tests.

4 Results

4.1 Selection of Embedding Models for Testing

Table 1 shows the accuracies of each document representation model combined with its own optimal classifier, before any augmentation (apart from lemmatisation, arguably) was applied. The *accuracy* of a model-classifier combination was computed as follows [1]:

$$accuracy = \frac{\sum_{c \in C} tp_c}{N} \tag{9}$$

where C is the set of all classes in the test, tp_c is the number of true positives for class c, i.e. instances of correct classification of documents belonging to it and N is the total number of documents classified. Each score shown here is an average of 1000 classification tests.

Table 1. Initial classification performance of the models with their optimal SVM classifiers.

Model	Accuracy	Optimal classifier
BOW-TF-IDF	84.0%	Linear SVC
Swivel MDV	80.0%	SVC with RBF kernel
Word2vec-SG MDV	76.8%	SVC with RBF kernel
Doc2vec-DBOW	74.2%	Linear SVC
GloVe MDV	68.6%	Linear SVC
FastText-SG MDV	66.6%	Linear SVC
Word2vec-CBOW MDV	65.1%	Linear SVC
Doc2vec-DM	63.6%	NuSVC with RBF kernel
FastText-CBOW MDV	59.9%	Linear SVC

Key: SVC: Support Vector Classifier, RBF: radial basis function

Since BOW-TF-IDF and doc2vec-DBOW were the two best-performing document vector models and Swivel and word2vec-skip-gram provided the best MDV, these were chosen for augmentation tests.

4.2 Weighting Method

In order to determine the optimal word vector weighting method out of the IDF, RF and Parallel IDF-CF weighting methods and MDV (where every token has the same weight), the performance of Swivel and word2vec-skip-gram MDV and WDV representations was tested on a variety of POS filters.

Table 2. POS filter results for the Swivel model. SO indicates the use of subject/object tagging. The peak accuracy (in bold) out of all tests in this paper was observed here.

Weighting	N	N+A	N+V	N+ADV	N+A+V	N+A+ADV	N+V+ADV	N+A+V+ADV
MDV	83.7%	84.1%	83.2%	83.8%	82.0%	83.9%	82.6%	82.7%
IDF WDV	83.7%	83.2%	83.2%	83.6%	82.2%	83.4%	82.5%	82.5%
RF WDV	83.5%	**84.1%**	83.5%	83.6%	82.5%	84.1%	83.3%	83.3%
ParIDF-CF WDV	76.6%	78.6%	77.1%	76.8%	77.2%	78.2%	77.4%	77.9%
MDV SO	83.4%	83.2%	83.0%	83.5%	82.4%	83.9%	83.3%	82.6%
IDF WDV SO	82.8%	83.2%	82.9%	83.3%	82.5%	83.2%	82.9%	82.4%
RF WDV SO	83.7%	83.3%	83.5%	83.9%	82.9%	83.7%	83.8%	82.9%
ParIDF-CF WDV SO	76.3%	77.8%	76.5%	75.9%	77.5%	77.9%	77.1%	77.3%

Tables 2 and 3 don't show an obvious winner among the weighting methods. Therefore, for each weighting method, all instances where the method was used were considered and the mean of the accuracies obtained in those instances was taken. These mean values can be seen in Table 4. Because IDF weighting gave the highest mean score, it was chosen as the optimal weighting method.

4.3 Evaluation of Augmentation Parameters

The results of augmentation parameter evaluations are presented in Tables 5 and 6. Due to the abundance of possible combinations between augmentation parameters and word-embedding models it was not possible to choose a single best result via direct Pareto elimination. Henceforth, in order to select a single best candidate for further analysis, the following loss function was used for evaluation.

$$Q_i = \sum_j \left[\frac{\max_i P_{i,j}}{P_{i,j}} - 1 \right], \tag{10}$$

where Q_i - the loss score of the ith augmentation dataset, $P_{i,j}$ - the accuracy score of the ith augmentation dataset with the jth document vector model.

Table 3. POS filter results for the word2vec model. SO indicates the use of subject/object tagging.

Weighting	N	N+A	N+V	N+ADV	N+A+V	N+A+ADV	N+V+ADV	N+A+V+ADV
SG MDV	78.4%	78.6%	77.4%	77.9%	77.1%	78.1%	76.6%	77.3%
SG-IDF WDV	78.6%	78.6%	78.0%	78.3%	77.7%	78.3%	77.4%	77.7%
SG-RF WDV	23.8%	21.9%	20.2%	20.7%	20.9%	19.3%	17.4%	21.1%
SG-Par IDF-CF WDV	72.7%	73.9%	73.7%	73.4%	74.2%	74.3%	73.5%	74.1%
SG MDV SO	77.3%	77.7%	78.1%	77.7%	78.4%	77.8%	78.8%	78.7%
SG-IDF WDV SO	81.1%	81.5%	81.7%	81.5%	81.8%	81.3%	82.3%	82.1%
SG-RF WDV SO	21.3%	23.9%	24.8%	20.5%	18.6%	20.8%	18.1%	21.3%
SG-Par IDF-CF WDV SO	75.1%	75.7%	76.1%	75.5%	76.0%	75.6%	76.0%	75.9%

Table 4. Mean accuracy by weighting method

Method	Mean accuracy
MDV	81.5%
IDF WDV	**82.5%**
RF WDV	62.7%
Parallel IDF-CF WDV	76.8%

The loss function is smaller when the relative loss of accuracy is small with respect to the best accuracy achieved among all augmentation parameters for a document vector model. The set of augmentation parameters that minimised this loss function was selected for the last step of analysis.

4.4 POS Filters and Tagging

Next, a POS filter to be used in the augmentation technique had to be chosen. At the same time, a decision had to be made on whether or not subject/object tagging should be used. These choices were made based on a loss score outlined in Subsect. 4.3, where the configuration with the smallest loss was chosen. Table 5 shows the performance of each document vector model when trained on differently POS-filtered versions of the corpus. Table 6 shows the same, but with subject- and/or object-tagged tokens added to the corpus.

Based on Tables 5 and 6, it is clear that a POS filter that includes nouns and adjectives only (plus special exceptions) is the best choice, as it records the lowest loss across all models regardless of tagging. The inclusion of subject/object tags also clearly improves the result as every thus tagged version of the corpus records a smaller overall loss than any non-tagged version.

Table 5. POS filter results without subject/object tagging. The lowest loss score is in bold.

Model	doc2vec DBOW	Swivel SG-IDF WDV	BOW TF-IDF	word2vec SG-IDF WDV	LOSS
N	72.7%	83.7%	83.1%	78.6%	10.2%
N+A	74.0%	83.2%	83.7%	78.6%	**8.1%**
N+V	72.5%	83.2%	83.0%	78.0%	12.0%
N+ADV	73.1%	83.6%	83.1%	78.3%	10.1%
N+A+V	74.3%	82.2%	83.6%	77.7%	10.3%
N+A+ADV	74.0%	83.4%	83.6%	78.3%	8.5%
N+V+ADV	72.4%	82.5%	83.0%	77.4%	13.8%
N+A+V+ADV	74.1%	82.5%	83.4%	77.7%	10.4%

Table 6. POS filter results with subject/object tagging. The lowest loss score is in bold.

Model	doc2vec DBOW	Swivel SG-IDF WDV	BOW TF-IDF	word2vec SG-IDF WDV	LOSS
N	74.8%	82.8%	82.8%	81.1%	5.4%
N+A	76.0%	83.2%	83.4%	81.5%	**2.2%**
N+V	75.0%	82.9%	83.0%	81.7%	4.2%
N+ADV	75.0%	83.3%	82.9%	81.5%	4.0%
N+A+V	76.2%	82.5%	83.4%	81.8%	2.4%
N+A+ADV	75.9%	83.2%	83.5%	81.3%	2.4%
N+V+ADV	75.0%	82.9%	82.9%	82.3%	3.5%
N+A+V+ADV	75.6%	82.4%	83.4%	82.1%	2.8%

4.5 Classification Results for Augmented Text

At this stage, there was a final augmentative configuration: POS weighting for WDV models and a SO-tagged corpus to which a nouns+adjectives POS filter had been applied. The same models that were tested in Subsect. 4.1 were trained with this configuration, yielding results outlined in Table 7. Each of these is also an average result of 1000 runs.

A substantial overall improvement, 4.1% points on average, can be observed from these results. Every model apart from the GloVe WDV model recorded a higher classification accuracy when the chosen augmentation technique was applied, compared to their performance before augmentation. The increase in the accuracy of the word2vec-Continuous Bag of Words (CBOW) model was large enough to enable it to surpass doc2vec-DBOW.

Table 7. Classification performance of the augmented models with their optimal SVM classifiers and the (Δ) in accuracy in terms of percentage points (pp).

Model	Accuracy	Δ	Optimal classifier
BOW-TF-IDF	83.4%	−0.6 pp	Linear SVC
Swivel-IDF WDV	83.1%	+3.1 pp	SVC with RBF kernel
Word2vec-SG-IDF WDV	81.3%	+4.5 pp	SVC with RBF kernel
Word2vec-CBOW-IDF WDV	76.5%	+11.4 pp	Linear SVC
Doc2vec-DBOW	76.1%	+1.9 pp	Linear SVC
FastText-SG-IDF WDV	74.9%	+8.3 pp	Linear SVC
Doc2vec-DM	71.6%	+8.0 pp	NuSVC with RBF kernel
FastText-CBOW-IDF WDV	69.0%	+9.1 pp	Linear SVC
GloVe-IDF WDV	59.3%	−9.3 pp	Linear SVC
Average Δ		**+4.1 pp**	

5 Analysis of the Impact of Augmentation Parameters

Although it could be argued that at least some of the improvement achieved through POS filtering may be mainly due to the stop-word filtering inherent in it, it is clear that POS filtering can improve classification accuracy on Russian-language text. This is evidenced not only by the final result, but also by the differing results between different POS filters. Previous studies, that investigated POS filtering on other languages, show that its effects vary by language: in [4], it was found to have a positive impact on the classification of Brazilian Portuguese emails. At the same time, results in [25] indicate that POS filtering largely worsens classification accuracy on Finnish social media texts. In [6], POS filtering was found to improve accuracy in Czech text classification. Interestingly, the results there are somewhat similar to those in this paper. Since Czech belongs to the same Slavic language family as Russian, this could indicate that similar techniques work on similar languages.

The subject/object-tagging (SO tagging) of words mostly improved document classification accuracy by adding information that would otherwise have been lost. While it did slightly worsen the results for Swivel and BOW-TF-IDF, the two best-performing models, it significantly improved them for doc2vec-DBOW and word2vec-skip-gram. The natural conclusion is that it is advisable to include subject/object tagging in a generic text augmentation technique, but that it may be best to leave it aside in the case of some specific document representation models. The tagged models can't be said to have benefited from their wider context windows compared to their respective untagged versions, because the increase (doubling) in the window size is approximately proportional to the increase (111%) in the size of the corpus when tagged tokens are added, or in fact, even slightly lower than that.

Among word weighting methods, IDF recorded the best overall performance, improving on MDV results. This is in line with known previous research on IDF word vector weighting [5], as well as the long-standing success of TF-IDF as a Bag-of-Words-type feature vector model [26]. It is worth noting, however, that the single best result was in fact recorded with an RF-weighted Swivel WDV trained on an untagged corpus, to which the winning N+A POS filter had been applied, at a 100-run mean accuracy of 84.1%. This would therefore be the optimal combination when choosing both augmentation parameters and a document representation model. However, RF's success as a weighting method for Swivel cannot be generalised to other word vector models, as it recorded very poor results with word2vec-skip-gram, with accuracies close to 20%, the random chance level for five classes.

6 Conclusion

In this work, the classification performance of nine different document vector representations of documents was tested with and without various augmentation techniques. Through these tests, to the authors' best knowledge, the first such analysis on Russian text was conducted.

It has been established that POS filtering as well as the use of IDF-weighted document vectors both improve classification accuracy on Russian documents. Subject/object tagging has also been introduced as another way to potentially enhance the semantic representativeness of a corpus. The augmentation technique formed as a result of the tests achieved an average increase of 4.1% points in terms of classification accuracy. This technique included the IDF weighting of word vectors, a noun+adjective POS filter, as well as subject/object tagging. The peak accuracy achieved in any test was 84.1%, which was achieved with the same filter but without tagging and with RF weighting instead of IDF.

All vector representations were computed using the whole corpus, which means that in any classification test, the documents represented by vectors in the test set will have contributed to the training of the training set, based on which the test set is then classified. While this would theoretically be a problem, making the results of the classification test ungeneralisable to the classification of external documents, it is important to recognise that most models can be trained further on new words and documents. Even when that isn't possible, a model can be regenerated using a corpus expanded with the new entries with no great cost, provided that the corpus isn't very large.

The results presented in this paper are specific to the news domain, since only news articles were used in classification tasks. While one still gets an indication on how the techniques used work on Russian, the combinations of techniques that work best on documents of other domains may differ from those that were found to be best here. Similarly, results obtained here may not necessarily be generalisable to tasks other than document classification. Future research applying the augmentation techniques used here to corpora representing other text domains, as well as to other semantic tasks, would therefore be highly desirable.

It would also be interesting to see whether or not these methods work with other languages, particularly other Slavic languages.

References

1. Van Asch, V.: Macro- and micro-averaged evaluation measures. Tech. Rep., (2013)
2. Bjerva, J., Plank, B., Bos, J.: Semantic tagging with deep residual networks. CoRR abs/1609.07053 (2016). http://arxiv.org/abs/1609.07053
3. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. CoRR abs/1607.04606 (2016). http://arxiv.org/abs/1607.04606
4. Bonatti, R., de Paula, A.G., Lamarca, V.S., Cozman, F.G.: Effect of part-of-speech and lemmatization filtering in email classification for automatic reply. In: AAAI Workshop: Knowledge Extraction from Text (2016)
5. Galke, L., Saleh, A., Scherp, A.: Word embeddings for practical information retrieval. In: Eibl, M., Gaedke, M. (eds.) INFORMATIK 2017, pp. 2155–2167. Gesellschaft für Informatik, Bonn (2017)
6. Hrala, M., Král, P.: Evaluation of the document classification approaches. In: Burduk, R., Jackowski, K., Kurzynski, M., Wozniak, M., Zolnierek, A. (eds.) CORES 2013. AISC, pp. 877–885. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-319-00969-8_86
7. Jones, K.S.: A statistical interpretation of term specificity and its application in retrieval. J. Doc. **28**, 11–21 (1972)
8. Korpusik, M., Huang, C., Price, M., Glass, J.: Distributional semantics for understanding spoken meal descriptions. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6070–6074 (2016). https://doi.org/10.1109/ICASSP.2016.7472843
9. Kutuzov, A., Andreev, I.: Texts in, meaning out: neural language models in semantic similarity task for Russian. ArXiv e-prints, April 2015
10. Lan, M., Tan, C.L., Su, J., Lu, Y.: Supervised and traditional term weighting methods for automatic text categorization. IEEE Trans. Pattern Anal. Mach. Intell. **31**(4), 721–735 (2009). https://doi.org/10.1109/TPAMI.2008.110
11. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. CoRR abs/1405.4053 (2014). http://arxiv.org/abs/1405.4053
12. Liu, M., Yang, J.: An improvement of TFIDF weighting in text categorization. In: International Conference on Computer Technology and Science (ICCTS), vol. 47, pp. 44–47 (2012). https://doi.org/10.7763/IPCSIT.2012.V47.9
13. Lyashevskaya, O., Bocharov, V., Sorokin, A., Shavrina, T., Granovsky, D., Alexeeva, S.: Text collections for evaluation of Russian morphological taggers. J. Linguist./Jazykovedný casopis **68**(2), 258–267 (2017)
14. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR abs/1301.3781 (2013). http://arxiv.org/abs/1301.3781
15. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. CoRR abs/1310.4546 (2013). http://arxiv.org/abs/1310.4546
16. Nguyen, D.Q., Nguyen, D.Q., Pham, D.D., Pham, S.B.: A robust transformation-based learning approach using ripple down rules for part-of-speech tagging. AI Commun. **29**(3), 409–422 (2016). https://doi.org/10.3233/AIC-150698

17. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
18. Pennington, J., Socher, R., Manning, C.D.: Glove: global vectors for word representation. In: Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014). http://www.aclweb.org/anthology/D14-1162
19. Řehůřek, R., Sojka, P.: Software framework for topic modelling with large corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, pp. 45–50. ELRA, Valletta, May 2010. http://is.muni.cz/publication/884893/en
20. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Inf. Process. Manag. **24**(5), 513–523 (1988). https://doi.org/10.1016/0306-4573(88)90021-0
21. Segalovich, I.: A fast morphological algorithm with unknown word guessing induced by a dictionary for a web search engine. In: MLMTA, pp. 273–280 (2003). http://cache-mskstoredata09.cdn.yandex.net/download.yandex.ru/company/iseg-las-vegas.pdf
22. Senel, L.K., Utlu, I., Yucesoy, V., Koc, A., Cukur, T.: Semantic structure and interpretability of word embeddings. IEEE/ACM Trans. Audio Speech Lang. Process., 1 (2018). https://doi.org/10.1109/TASLP.2018.2837384
23. Shazeer, N., Doherty, R., Evans, C., Waterson, C.: Swivel: improving embeddings by noticing what's missing. CoRR abs/1602.02215 (2016). http://arxiv.org/abs/1602.02215
24. Toman, M., Tesar, R., Jezek, K.: Influence of word normalization on text classification. In: Proceedings of InSciT, pp. 354–358 (2006). http://www.kiv.zcu.cz/research/groups/text/publications/inscit20060710.pdf
25. Venekoski, V., Puuska, S., Vankka, J.: Vector space representations of documents in classifying finnish social media texts. In: Dregvaite, G., Damasevicius, R. (eds.) ICIST 2016. CCIS, vol. 639, pp. 525–535. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46254-7_42
26. Wang, S., Manning, C.D.: Baselines and bigrams: simple, good sentiment and topic classification. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers, ACL 2012, vol. 2, pp. 90–94. Association for Computational Linguistics, Stroudsburg (2012). http://dl.acm.org/citation.cfm?id=2390665.2390688

Accounting for Named Entities in Intent Recognition from Short Chats

Ghislain Landry Tsafack(✉) and Sharva Kant

Centrica IS Data Science, Lovett Road, Staines, TW18 3AR, UK
ghislain.tsafack@centrica.com, sharva.kant@britishgas.co.uk

Abstract. The operational cost of call centres accounts for a large part of the total spending of any modern organization. As a consequence, the automated conversation agent powered by Artificial Intelligence (AI) through Natural Language Processing (NLP) alternative has gained major attraction over the past years. Efforts to achieve such level of automation generally rely on predefined business intents or intents, which are in turn tightly related to business entities or processes that they represent. As the success of the automated conversation agent fully relies on its ability to accurately recognise user intents, a good automated agent will be the one that recognises intents it is meant to recognise. In light of the strong relationship that exists between business entities and business intents or entities and intents in general, we propose two approaches for accounting for named entities in the task of intents recognition from short chats. The first approach relies on Bi-Normal Separation (BNS) to weight term features that are named entities more than other features, whereas, the second approach takes advantage of word embedding to encode the relationship between entities and chats. Evaluation of proposed methodologies, on a data set composed of one to one conversations between human actors, suggests that accounting for named entities improves the performance of the intents recognition task.

Keywords: Natural Language Processing · Named entity recognition Features selection · Intent recognition

1 Introduction

Over the past recent years, modern organizations have been devoting tremendous efforts to reduce the load of their call centres. Most recent improvement towards that direction involve incentivizing customers to use online communication channels, where their interaction with the organisation could be handled by an automated web-chat interface powered by Artificial Intelligence (AI), whenever possible. This is very cost effective and more importantly reduces the need for a phone call (along with long and frustrating hold times) or chat-based discussion with a person. It is therefore not surprising that Natural Language Processing (NLP) or text analysis is becoming a key player in the mainstream of customer service.

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, pp. 587–596, 2018.
https://doi.org/10.1007/978-3-319-99972-2_48

However, the ability to accurately identify users intents from web-chat texts hinders NLP success in such a sensitive area as users are often left in frustration by the low precision and recall of the underlying intents recognition engine. Intents detection is without loss of generality, a form of document classification. In such settings, documents are short texts, often highly contextual, sent by customers either as a query or in response to the web-chat. Whereas classes of documents are predefined domain specific intentions [1]. Similar to Internet documents where text, hyper-links, meta-data, and other multimedia are used for classification [2–4]; chat texts contain contextual information such as products' references and industry specific keywords that can be used to create a supervised automatic classifier by means of machine learning algorithms. Above mentioned machine learning algorithms often expect fixed size numeric vectors as input. Consequently, input texts, usually of variable length, need to be transformed into representative fixed size numeric vectors: this process is known as feature extraction.

In a commercial environment, the term "business intent" is often used to refer to intents that directly map the business. For example, if it is a Product information system, then an utterance from the customer like "Is my boiler and controls service due this week?" is a business intent that intends to find out the date of the next home care visit for set customer and should be labelled accordingly with an understandable name like "HomeCareVisitByDate". In this context, "home care" is one of the products offered by British Gas. Business intents have metadata about the intent also referred to as "Entities". For example, for an intent "HomeCareVisitByDate" – Sample utterance "Is my boiler and controls visit due this week?". Here boiler and controls service is the name of one of the many home care services, for which the user "intends" to find out when it is going to take place. In addition, the term "week" expresses time and will generally be replaced by the mention "time" when training the intents recognition classifier.

This paper proposes two approaches for accounting for named entities, both business and standard entities, in the context of intents recognition. The first approach relies on Bi-Normal Separation (BNS) to increase the scale of named entity term features. The second approach suggests supplementing term features with named entity embedded features. Bi-Normal Separation (BNS)[7] is a feature selection approach best suited for short text classification. BNS is known to utilise the fact that some keywords can also determine the absence of some classes and is often perceived as a better metric for attaining best performance for accuracy, recall and F-score.

The remainder of this paper is organized as follows: Sect. 2 provides an overview of research work related to our contribution. Our methodology is introduced in Sect. 3. Experimental results are presented and analysed in Sect. 4. Finally, we conclude the paper and outline future directions in Sect. 5.

2 Related Work

We survey resources most related to our contribution through a brief review of named entity recognition, existing feature selection methods for text classification, and some of the most recent techniques used for intention recognition.

2.1 Named Entity Recognition

The majority of advanced techniques for extracting named entities from texts relies on machine learning techniques. [10] is probably one of the first attempts to systematically identify named entities from a text. To accomplish this authors rely on Hidden Markov Model (HMM), which is effective in other areas for NLP including Part-Of-Speech (POS) tagging. In [11], authors propose a much more complex approach for the task of named entity recognition through a combination of classifiers including Hidden Markov Model, Maximum Entropy, and Robust Linear Classifier. Authors of [9] explore the use of Conditional Random Fields (CRF) for various NLP tasks including NER. [12] introduces the concept of BILOU (Beginning - Inside - Last - Outside - Unit) for chunking, adding additional dimensions to the task of NER recognition. Above approaches have in common the fact that they use shallow parsing either through POS tagging or chunking. More recently, advances in the field of machine learning along with the availability of more powerful computers have enable the use of complex algorithms for the task of named entity recognition. Initial efforts in that direction replace words with embeddings using deep learning strategies, which minimize the need of previous costly features [13,14]. Latest advances in the field of NER involve the use of convolutional neural network (CNN) models [15,16]. spaCy's[1] named entity recognition module is an example implementation of CNN for named entity recognition and has been shown to outperform other state-of-art benchmarks in the task of entity recognition [17]. Although name entity recognition has been extensively studied, to the best of our knowledge, little to no effort has been devoted to investigate their impact on intents recognition in a commercial environment, where they can be thought of as meta data and often provide significant information on users' intentions. For example, consider the text "I lost my meter's key". It is essential to understand that "meter" in this context is a business "entity" referring to the electricity or gas counter rather than the base unit of length.

2.2 Machine Learning

A large number of machine learning algorithms are available for the task of text classification. These include, but are not limited to, Naive Bayes, Maximum Entropy, K-Nearest Neighbour (K-NN), and Support Vector Machine (SVM). The literature consistently reports that SVM performs better than the others [21, 22]. Hence, SVM is one of the algorithms we consider in our analysis. However,

[1] https://spacy.io.

commercial NLP as a service often require users to quickly train or (retrain) their intention recognition model, so we also investigate Random Forest (RF) [23], which has emerged as an efficient classification algorithm for a vast majority of machine learning problems. While SVM operates via identifying a boundary between different categories (often linear SVM when the boundary is linear), the basic concept of RF is the construction of a set of decision trees. It offers the advantage of being faster and less computationally expensive than SVM. Although Deep Neural Networks are becoming a promising alternative to state-of-the art machine learning algorithms; they are time consuming and might be impractical in the context of NLP as a service, especially given that users are left with the task of training and optimizing their own model. In addition, they typically perform badly on small amounts of data [24]. Hence, we do not investigate the use of deep neural networks in this document. One could argue the use of word embeddings, however, it has two major caveats. One of those caveat being the fact that its use requires practitioner to either train context specific word embeddings from the text available or use global dictionaries. Although, this might be seen as a caveat, it can be exploited to create embedding that encode specific relationships, we expand on this idea in Sect. 3.

Global dictionaries are trained on large datasets of global values and meanings. So, retraining them is often required to incorporate the contextual meaning of words. The challenge of re-training is debatable as to how much of data is required to re-train the dense layers of word embeddings. Another issue with using global vectors are appearance of new contextual word vectors in an industrial setup. For example, an abbreviation or a pronoun that is used only in the context of the company could not have been possibly included in the global dataset no matter how big or from where it has been taken. With limited knowledge of deep learning and the meaning of dimensions it throws out from training, how can we then possibly initialize these vectors? As for the new vector, the entity needs to be re-trained with the presence of other word vectors who themselves may not be perfect for the context. The situation is even more complicated when there are multiple words of this nature who have their word embeddings missing [25].

3 Methodology

3.1 BNS Based Approach

In this setting, we use an approach similar to the one discussed in [4], where documents are arranged in a term document matrix A such that $A_{i,j}$ is the count of the term i in document j. When extracting features for a language model, it is standard practice to remove words commonly referred to as stop words. Briefly, stop words are very frequent natural language words that have little meaning, such as "and", "the", "to", and similar words. In addition, we remove words with very high frequency as they usually convey little information. For example, in customer support chats about smart meter installation and related services, the term "smart meter" (meter) is likely to appear at a very high frequency, but conveys little to no information about customers' intentions.

The term documents matrix gives more weight to longer documents than shorter documents. This is addressed by using term frequencies rather than actual counts. Finally, the *tf-idf* (term frequency times inverse document frequency) is derived from the term frequency matrix. The *tf-idf* has the effect of reducing the weight of more common words. Our *tf-idf* is generated using the Python scikit-learn[2] library which defines IDF as:

$$idf(t) = \log(\frac{n_d}{df(d,t)}) + 1$$

Where n_d is the number of documents, and d and t are respectively document and term.

We define a named entity document of a document D as a sequence of named entities found in D. Using this definition, we create a named entity document corpus from the training set. BNS is then applied to the above mentioned named entity document corpus in order to compute the BNS score of each named entity. We then construct the feature matrix from the *tf-idf* using Eq. 1, where NE stands for named entity. A similar approach using BNS is described in [7], however, we do not rely on binary features because the BNS score for word features in this setting are all less than one, in which case applying BNS scaling makes no difference. Named entities are extracted using Spacy (version 2), which has an accuracy of about 92.6%[3]. Equation 1 has the effect of increasing the range of word features of interest, therefore allowing us to encode our domain knowledge into the classifier.

$$tf\text{-}idf_{(i,j)} = \begin{cases} 0\ if\ tf\text{-}idf_{(i,j)}\ is\ 0, \\ \\ tf\text{-}idf_{(i,j)} + BNS(term_i)\ if\ NE(term_i) \\ tf\text{-}idf_{(i,j)}\ otherwise. \end{cases} \quad (1)$$

where $BNS(term_i)$ is the bi-normal separation score of $term_i$ and is computed as described in [7].

At the one hand, $tf\text{-}idf_{(i,j)}$ is biased towards words with high frequency, but overall rare in the collection. At the other hand, BNS is biased towards words with high category ratio. Consequently, Eq. 1 combines those two properties to weight named entities (could be any term), in documents where they are frequent.

3.2 Word Embedding Approach

Our effort to account for named entities when performing the task of Intents recognition, led us to Pointwise Mutual Information (PMI). Formally, it is a measure of how often two events occur, compared with what we would expect if

[2] http://scikit-learn.org/stable/modules/feature_extraction.html.
[3] https://spacy.io/usage/facts-figures.

they were independent. PMI is a commonly used vector representation of a word, it is used to compute the term context matrix representation of a collection of documents. Such matrix is often referred to as PMI-matrix. Unlike the term-document matrix, in the PMI matrix, columns are labelled by words rather than documents and each cell records the PMI of the row word (target) and the column word (context).

Using the PMI-matrix, we create an embedding of each document in our corpus as described by Eq. 2. As the result of Eq. 2 each document in the corpus is encoded by their relationship with entities.

$$E_{(i,j)} = \begin{cases} \max_{x\, \in \{NE\}} PMI(x\ , w_j)\ \forall\ w_j\ \in\ document_i \\ \\ where,\ NE\ stands\ for\ named\ entity \\ and\ w_j,\ the\ word\ at\ position\ j\ in\ document_i \end{cases} \tag{2}$$

Finally, the feature set constructed from Eq. 2, which we refer to as "entity embedded features", is combined with term features as described by Eq. 3 to form the final feature matrix M. When α is approaching zero, variance is dominated by embedded features whereas term features dominate when α is approaching 1.

$$M = \begin{bmatrix} \alpha\, tf\text{-}idf \\ \\ (1-\alpha)\ E \end{bmatrix} \tag{3}$$

4 Experimental Results and Discussion

This section presents and discusses preliminary results of our implementation of the methodology discussed in Sect. 3. Different configuration scenarios are considered in order to investigate and gain a better understanding of the impact of proposed methodologies on Intent recognition. We start by describing the data that we use, we then follow with a discussion of our experimental results.

4.1 The Data

Our data set is made up of 600 one to one chat conversations between human actors, whom we refer to as agents and visitors, about prepayment smart meters installation and related enquiries. Automatically labelling chat conversations is a complex and error prone procedure. Consequently, we manually labelled visitors' intentions into three set of intentions or categories of almost equal proportions: *KeyRelatedProblem* (200), *CreditNotShowingUp* (189), and *TopUpOptions* (211). All personal information including postcodes, phone numbers and email addresses are removed. Resulting texts containing users' intentions have 116 characters on average. We further split the data set into training (70%) and testing (30%) sets.

Prepayment smart meters are electric and/or gas metering devices, to which is associated a card (called key) that customers use to credit it, offering customers a fine grained control of their energy consumption. It appears that above mentioned categories are strongly related to our internal processes. For example, *KeyRelatedProblem* include enquiries about the meter key, such as key loss, broken key, etc.

4.2 Results and Discussion

This section aims at understanding the impact of complementing the traditional bag-of-word model with a bag of named entities. We use linear SVM and random forest implementations provided by the Python scikit-learn library. Tuning parameters for both algorithms are left to their default values, with the exception of the tree parameter for random forest. As performance evaluation measure, we use the F_1 score; the harmonic average of the precision and recall defined as:

$$F_1 = 2 \cdot \frac{(precision \times recall)}{(precision + recall)}$$

A good F_1 score requires both a good precision and a good recall on the positive class. The overall F_1 is computed as the micro-average of per-class F_1 score of each of the three classes available. As opposed to the macro-average, the micro-average provides a better sense of the overall accuracy as it weights individual F_1 scores according to the number of documents that happen to be available in each class.

Figure 1 shows the average difference in F_1 score for both random forest and support vector machine under different feature models. The prefix "ext-" indicates that the input feature matrix is computed using the methodology we described in Sect. 3. We can observe a up to 2% improvement in F_1 score depending on the classifier. This suggests that weighting named entities slightly more heavily than other features improves performance of the intents recognition task. This can be attributed to the fact that the dot-product distance as evaluated by the linear SVM kernel is affected by the scale of cases involved [8].

As mentioned in Sect. 3, we compute named entities BNS scores from the named entity document corpus built from the training set. Alternatively, scores can be computed from the original training set, we do not show results for such scenario because it gives similar performance for SVM and no improvement for

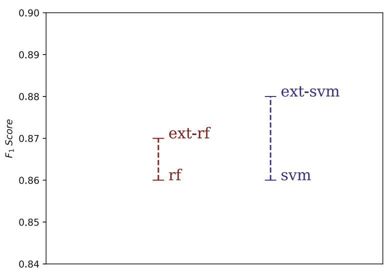


Fig. 1. Average difference in F_1 score between the scenario where term features are used and the scenario entity term features are given more weight. Considered machine learning algorithms are SVM and RF.

random forest. Figure 2, where the y-axis represents the F_1 score and the x-axis the number of features, shows the performance of random forest and support vector machine with respect to the number of features. On the one hand, random forest with standard $tf\text{-}idf_{(i,j)}$, RF on the graph of Fig. 2 seems to perform better as the number of features increases. On the other hand SVM with named entities given more weight (ext-svm) consistently performs better than SVM.

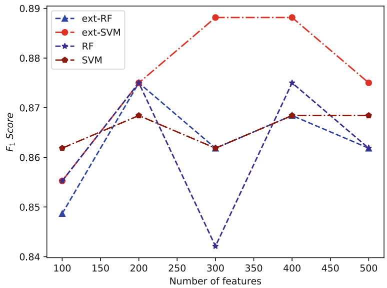


Fig. 2. F_1 score for random forest and support vector machine with respect to the number of features.

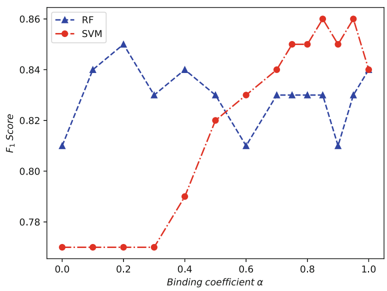


Fig. 3. Variation of the F_1 score for random forest and support vector machine with respect to the the binding coefficient α.

The effect of using embedding to complement term features is presented in Fig. 3. More explicitly, Fig. 3 where the x-axis represents different values of the binding coefficient α and the y-axis represents F_1 score, shows the variation of the F_1 score for random forest and support vector machine with respect to the binding factor α when the entire feature set is used. We can observe that the F_1 score of term features, when $\alpha = 1$ is improved by adding entity embedded features. The graph of Fig. 3 also suggests that the correct binding factor might depend on the algorithm at hand, so, we recommend trying multiple values preferably on a validation set. As a general observation, random forest seems to perform better with lower values of α, reaching its maximum F_1 when $\alpha = 0.2$. Whereas SVM in contrary seems to favour higher values of aplha, reaching its highest F_1 score when $\alpha = 0.85$.

5 Conclusion and Future Directions

In this paper, we present two approaches for enriching the task of intents recognition with our knowledge of named entities. While the first approach relies on Bi-Normal Separation feature scoring method to assign more weights to term features that are named entities. The second approach consists of supplementing term features with named entity embedded features.

We investigate the effectiveness of the proposed methodologies by evaluating them on a chat data set composed of one to one conversation between human actors. Comparison with baseline feature selection, where no emphasis is put on named entities shows the benefits of this approach depends on the machine learning algorithm as well the size of the feature set. However, we notice an average improvement of 2% of the F_1 score for support vector machine. The second approach exhibits a clear synergy between term features and entity embedded features and suggests that the task of intents classification can benefit from such entity embedded features. Although, we specifically targeted named entities, we believe this approach can easily be extended to any feature in the context of Intents recognition. Future direction include investigating the effectiveness of the proposed methodology on different data sets as well as less efficient algorithms.

References

1. Aixin, S.: Short text classification using very few words. In: Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1145–1146. AMC, Portland (2012)
2. T. Kolenda, L. Hansen, J. Larsen and O. Winther, Independent component analysis for understanding multimedia content. In: Bengio, S.,. Larsen, J., Bourlard, H., Adali, T., Douglas, S. (eds.) Proceedings of IEEE Workshop on Neural Networks for Signal Processing XII, pp. 757–766. IEEE Press, Piscataway (2002)
3. Kosala, R., Blockeel, H.: Web mining research: a survey. In: SIGKDD: SIGKDD Explorations: Newsletter of the Special Interest Group (SIG) on Knowledge Discovery & Data Mining, pp. 1–15. ACM Press, ACM (2000)
4. Madsen, R.E., Larsen, J., Hansen, L.K.: Part-of-speech enhanced context recognition. In: Proceedings of the 14th IEEE Signal Processing Society Workshop in Machine Learning for Signal Processing, pp. 635–643 (2004)
5. Deerwester, S., Dumais, S., Landauer, T., Funas, G., Harshman, R.: Indexing by latent semantic analysis. J. Am. Soc. Inf. Sci. **41**, 391–407 (1990)
6. Furnas, G.W., Deerwester, S., Dumais, S.T., Landauer, T.K., Harshman, R.A., Streeter, L.A., Lochbaum, K.E.: Information retrieval using a singular value decomposition model of latent semantic structure. In: Proceedings of the 11th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Grenoble, France. ACM (1988)
7. Forman, G.: An extensive empirical study of feature selection metrics for text classification. J. Mach. Learn. Res. **3**, 1289–1305 (2003)
8. Joachims, T.: Text categorization with support vector machines: learning with many relevant features. In: Proceedings of the 10th European Conference on Machine Learning (ECML), Berlin, pp. 137–142 (1998)

9. McCallum, A., Wei, L.: Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In: Proceedings of the 7th Conference on Natural Language Learning, Edmonton, Canada, pp. 188–191 (2003)
10. Bikel, M.D., Schwartz, R., Weischedel, R.M.: An algorithm that learns what's in a name. Mach. Learn. J. **34**(1–3), 211–231 (1999)
11. Radu, F., Ittycheriah, A., Hongyan, J., Tong, Z.: Named entity recognition through classifier combination. In: Proceedings of the 7th Conference on Natural Language Learning, Edmonton, Canada, pp. 168–171 (2003)
12. Ratinov, L., Roth, D.: Design challenges and misconceptions in named entity recognition. In: Proceedings of the 13th Conference on Computational Natural Language Learning, Boulder, Colorado, pp. 147–155 (2009)
13. Ronan, C., Jason, W., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. J. Mach. Learn. Res. **12**, 2493–2537 (2011)
14. Passos, A., Kumar, V., McCallum, A.: Lexicon infused phrase embeddings for named entity resolution. In: CoRR (2014)
15. Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.: Neural architectures for named entity recognition. In: CoRR (2016)
16. Jason, P., Chiu, C., Nichols, E.: Named entity recognition with bidirectional LSTM-CNNs. In: CoRR (2016)
17. Choi, J.D., Tetreault, R.J., Stent, A.: It depends: dependency parser comparison using a web-based evaluation tool. In: Proceedings of Association for Computational Linguistics (ACL), pp. 387–396 (2015)
18. Shoushan, L., Rui, X., Chengqing, Z., Chu-Ren, H.: A framework of feature selection methods for text categorization. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP, vol. 2, pp. 692–700 (2009)
19. Dhillon, I.S., Subramanyam, M., Rahul, K.: A divisive information theoretic feature clustering algorithm for text classification. J. Mach. Learn. Res. **3**, 1265–1287 (2003)
20. Evgeniy, G., Shaul, M.: Text categorization with many redundant features: using aggressive feature selection to make SVMs competitive with C4.5. In: Proceedings of the 21st International Conference on Machine Learning, Banff, Alberta, Canada, pp. 2493–2537 (2004)
21. Yang, Y., Pedersen, J.O.: A comparative study on feature selection in text categorization. In: Proceedings of the Fourteenth International Conference on Machine Learning, San Francisco, CA, USA (1997)
22. Yang, Y., Liu, X.: A re-examination of text categorization methods. In: Proceedings of SIGIR-99, The 22nd Annual International ACM Conference on Research and Development in Information Retrieval (1999)
23. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
24. Hellrich, J., Hahn, U.: Bad company - neighborhoods in neural embedding spaces considered harmful. In: COLING (2016)
25. Li, J., Jurafsky, D.: Do multi-sense embeddings improve natural language understanding? In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, 17–21 September 2015

Correction to: Application of SEO Metrics to Determine the Quality of Wikipedia Articles and Their Sources

Włodzimierz Lewoniewski, Ralf-Christian Härting, Krzysztof Węcel, Christopher Reichstein, and Witold Abramowicz

Correction to:
Chapter "Application of SEO Metrics to Determine the Quality of Wikipedia Articles and Their Sources" in: R. Damaševičius and G. Vasiljevienė (Eds.): *Information and Software Technologies*, CCIS 920,
https://doi.org/10.1007/978-3-319-99972-2_11

The name of an author was spelled incorrectly in the originally published chapter. This was corrected.

The updated online version of this chapter can be found at
https://doi.org/10.1007/978-3-319-99972-2_11

R. Damaševičius and G. Vasiljevienė (Eds.): ICIST 2018, CCIS 920, p. E1, 2018.
https://doi.org/10.1007/978-3-319-99972-2_49

Author Index